

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、誤りがな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。

ユーザズ・マニュアル

保守/廃止

μSAP77016-B10

音声認識ミドルウェア

対象デバイス

μPD77110

μPD77111

μPD77112

μPD77113

μPD77113A

μPD77114

μPD77115

[メ モ]

目次要約

第 1 章	概 説	...	12
第 2 章	ライブラリ仕様	...	18
第 3 章	インストレーション	...	34
第 4 章	PC 版英語単語データ作成ツール	...	36
第 5 章	システム例	...	44
付 録	認識処理サンプル・ソース (sample.asm)	...	49

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。

巻末にアンケート・コーナを設けております。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 このマニュアルは、 μ PD77016 ファミリの応用システムを設計、開発するユーザを対象としています。

μ PD77016 ファミリは、 μ PD7701 x 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。
ただし、このマニュアルでは、 μ PD77110, 77111, 77112, 77113, 77113A, 77114, 77115 を対象デバイスにしています。

目的 このユーザズ・マニュアルは、 μ PD77016 ファミリの応用システムを設計、開発する際にサポートするミドルウェアを、ユーザに理解していただくことを目的としています。

構成 このユーザズ・マニュアルは、大きく分けて次の内容で構成されています。

- 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
- 第4章 PC 版英語単語データ作成ツール
- 第5章 システム例
- 付 録 認識処理サンプル・ソース (sample.asm)

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

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 ユーザズ・マニュアル** **命令編**を参照してください。

- 凡 例 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 ... x x x x または 0b x x x x
- : 10進数 ... x x x x
- : 16進数 ... 0x x x x x

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。

μPD77016 ファミリに関する資料

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 PD77110	U12395J	U12801J	U14623J	U13116J	U11958J	U12021J
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	
CC77016	ユーザーズ・マニュアル	U15037J	
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 ...	12
1.1 ミドルウェア ...	12
1.2 音声認識ミドルウェア ...	12
1.2.1 特 徴 ...	12
1.2.2 処理概要 ...	13
1.3 製品概要 ...	14
1.3.1 仕 様 ...	14
1.3.2 動作対象 DSP ...	14
1.3.3 必要メモリ・サイズ ...	15
1.3.4 ソフトウェア・ツール (Windows™版) ...	15
1.3.5 性 能 ...	16
1.3.6 パッケージ内容 ...	17
第2章 ライブラリ仕様 ...	18
2.1 ライブラリ概要 ...	18
2.2 アプリケーション・フロー ...	19
2.3 タイミング・ダイアグラム ...	20
2.4 メモリ・マップ ...	21
2.5 関数仕様 ...	22
2.5.1 vrg_InitDic 関数 ...	22
2.5.2 vrg_AddDic 関数 ...	23
2.5.3 vrg_DelDic 関数 ...	28
2.5.4 vrg_Init 関数 ...	29
2.5.5 vrg_Recog 関数 ...	31
2.5.6 vrg_GetVersion 関数 ...	33
第3章 インストレーション ...	34
3.1 インストレーション手順 ...	34
3.2 サンプル・プログラム作成手順 ...	34
3.3 シンボル名規約 ...	35
第4章 PC 版英語単語データ作成ツール ...	36
4.1 動作環境 ...	36
4.2 単語データ作成方法 ...	36
4.3 入力ファイル ...	36
4.3.1 音素記号一覧 ...	37
4.3.2 破裂音の記法 ...	37
4.3.3 音素記号 ...	38

4.3.4 Merriam Webster から ASCII 発音記号変換ルールについて (補足) ...	40
4.4 出力ファイル ...	41
4.5 エラー・コード ...	42
4.6 ワーニング ...	43
第5章 システム例 ...	44
5.1 タイミング・ファイルを使用したシミュレーション環境 ...	44
5.2 操作方法 ...	44
5.3 動作説明 ...	44
5.3.1 単語データの入力 (16 ビット・データ) ...	46
5.3.2 音声データの入力 (16 ビット・データ) ...	47
5.3.3 認識結果の出力 ...	48
付録 認識処理サンプル・ソース (sample.asm) ...	49

図の目次

図番号	タイトル, ページ
1-1	単語音声認識構成 ... 13
2-1	シングル・タスク・アプリケーション処理フロー例 ... 19
2-2	タイミング・ダイアグラム ... 20
2-3	メモリ・マップ例 ... 21
2-4	登録単語文字列格納イメージ (5文字の場合) ... 24
2-5	辞書格納領域 (1ワード単位) ... 27
2-6	vrg_result (1ワード単位) ... 32

表の目次

表番号	タイトル, ページ
1-1	音声認識ミドルウェア仕様 ... 14
1-2	必要メモリ・サイズ ... 15
1-3	1フレーム (16 ms) の認識処理に必要なサイクル数と MIPS 値 ... 16
1-4	最大認識単語数 ... 16
2-1	ライブラリ関数一覧 ... 18
2-2	vrg_InitDic 関数戻り値一覧 ... 22
2-3	vrg_AddDic 関数戻り値一覧 ... 23
2-4	登録単語文字列使用可能コード一覧 ... 25
2-5	使用可能文字一覧表 ... 26
2-6	単語 ID 列 ... 27
2-7	vrg_DelDic 関数戻り値一覧 ... 28
2-8	vrg_Init 関数戻り値一覧 ... 29
2-9	vrg_Recog 関数戻り値 R0 一覧 ... 31
3-1	シンボル名 ... 35
4-1	PC 版英語単語データ作成ツール引き数 ... 36
4-2	母音 (Vowels) ... 38
4-3	子音 (Consonants) ... 39
4-4	IPA と Merriam Webster の発音記号対応 ... 40
4-5	ストレス位置 ... 40
4-6	エラー・コード一覧 ... 42
4-7	ワーニング一覧 ... 43

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最大限に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 ファミリー用に，ヒューマン・マシン・インタフェースおよび信号処理技術をミドルウェアの形で提供しています。さまざまなユーザのニーズに対応して，優れたシステム・ソリューションを提供しています。

備考 DSP: Digital Signal Processor

1.2 音声認識ミドルウェア

発声された音声を認識する機能をもつミドルウェアです。

あらかじめ辞書に登録された単語の中で，発声された音声に最も近いと推定された単語の単語番号を認識結果として出力します。

1.2.1 特 徴

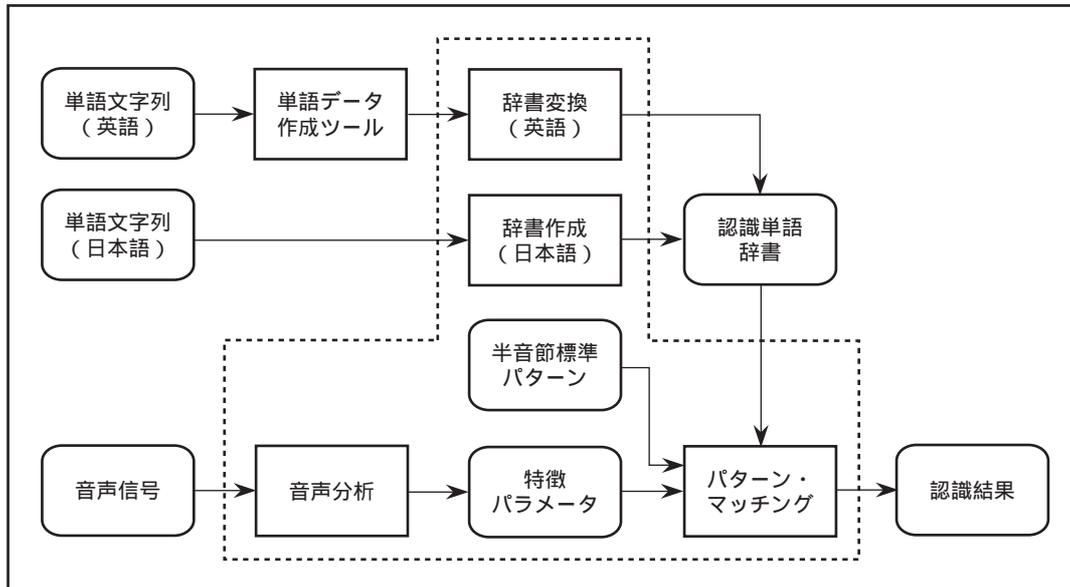
- 離散単語認識（単語単位で発声された言葉を認識）
- 事前発声不要な不特定話者単語認識
- 認識単語を全角かな文字により登録するため，認識辞書が簡単に作成可能（日本語のみ）[※]
- リアルタイム（発声終了後 0.4 秒で認識結果出力）での認識処理が可能

注 英語版では，テキストで記述した特殊発音記号を PC 上で変換したものを認識単語として登録します。

1.2.2 処理概要

音声認識の構成を次に示します。

図 1-1 単語音声認識構成



備考 破線内をミドルウェアとして提供します。

(1) 認識処理

(a) 音声信号入力

発声した音声を PCM 信号 (16 ビット, 8 kHz サンプリング) として入力します。

(b) 音声分析

入力された音声信号について、音声分析処理をして特徴パラメータを抽出します。このとき、周囲の定常ノイズによる影響を低減させる処理を行います。

(c) パターン・マッチング

音声分析処理により抽出された特徴パラメータに対し、あらかじめ用意した半音節単位の標準パターンと認識単語辞書を使用してパターン・マッチングを行います。

(d) 認識結果出力

パターン・マッチングの結果、入力した音声に似ていると推定できる単語を認識結果とします。認識結果として単語 ID とスコアを出力します。候補を最大 5 位まで出力します。単語 ID とスコアについては、2.5.5 `vrg_Recog` 関数を参照してください。

(2) 辞書作成機能

音声認識処理を行うときに使用する辞書を作成します。

日本語単語文字列をミドルウェアへ入力すると音声認識処理に必要な形式（認識単語辞書）に変換され、ユーザの確保した領域に格納されます。

英語版では、PC で動作する単語データ作成ツールを用いてミドルウェアへ入力する単語データを作成します。単語データ作成ツールについては、第 4 章 PC 版英語単語データ作成ツールを参照してください。

1.3 製品概要

1.3.1 仕 様

表 1-1 音声認識ミドルウェア仕様

項 目	仕 様
音声認識方式	半音節音声認識方式
認識対象言語	単語単位で発声された日本語の音声または英語の音声
認識対象話者	不特定話者（事前の学習，話者登録不要）
外部ノイズ対策	定常ノイズ・キャンセル
入力音声	16 ビット・リニア PCM サンプリング周波数：8.0 kHz 帯域：0.3～3.4 kHz 入力：1 ch
認識単語数	70～260（対象 DSP によって変わります）

注意 日本語音声認識と英語音声認識はそれぞれ別のライブラリとなっています。

認識率には個人差があります。

ノイズが大きいと十分な認識率が得られない場合があります。また、口元からマイクロホンまでの距離が遠い場合にも、音声に対して相対的にノイズが大きくなります。口元からマイクロホンまでの距離は15 cm以内を目安にしてください。

英語の音声は、標準的な米国人の発音を対象にしていますが、発音の仕方によっては十分な認識率を得られない場合があります。

1.3.2 動作対象 DSP

- 日本語版ライブラリ：μ PD77110, 77111, 77112, , 77113, 77113A, 77114, 77115
- 日本語耐雑音強化版ライブラリ：μ PD77110, 77113, 77113A, 77114
- 英語版ライブラリ：μ PD77113, 77113A, 77114

1.3.3 必要メモリ・サイズ

表 1-2 必要メモリ・サイズ

メモリ	種別	必要サイズ		
		日本語耐雑音強化版	日本語版	英語版
命令メモリ	-	6.4 (Kワード)	5.9 (Kワード)	
Xメモリ	RAM (可変) [※]	11.0 (ワード/単語) + 9.0 (ワード)		
	RAM (固定)	2.8 (Kワード)	2.2 (Kワード)	3.2 (Kワード)
	ROM	16.5 (Kワード)	12.1 (Kワード)	23.5 (Kワード)
Yメモリ	RAM (可変) [※]	31.0 (ワード/単語) + 2.0 (ワード)		
	RAM (固定)	2.0 (Kワード)	1.9 (Kワード)	
	ROM	12.6 (Kワード)	11.9 (Kワード)	19.1 (Kワード)

注 RAM (可変) は、認識単語数に依存して必要サイズが変化する領域です。

上記の値は、認識候補単語として 1 単語につき 5 文字の単語を使用した場合の目安です。英語の場合は、1 単語につき 10 音素の単語を使用した場合の目安です。

注意 命令メモリの1ワードは、32ビットです。Xメモリ、Yメモリの1ワードは、16ビットです。

1.3.4 ソフトウェア・ツール (Windows™ 版)

DSP ツール： WB77016 (ワークベンチ)

HSM77016 (ハイスピード・シミュレータ)

1.3.5 性 能

【1単語を辞書に登録するために必要なサイクル数】

日本語： 8248 サイクル

英語： 341 サイクル

備考 上記の値は，日本語は1単語5文字，英語は1単語10音素の場合です。

【1フレーム（16ms）の認識処理に必要なサイクル数とMIPS値】

表 1-3 1フレーム（16ms）の認識処理に必要なサイクル数とMIPS値

対象	音声検出状態	辞書単語数	最大[cycle]	最小[cycle]	MIPS ^注
日本語 耐雑音 強化版	音声未検出時	50 単語	63525	703	3.98
		100 単語	63525	703	3.98
	音声検出時	50 単語	153802	145499	9.62
		100 単語	167792	156350	10.49
英語版	音声未検出時	50 単語	45398	703	2.84
		100 単語	45398	703	2.84
	音声検出時	50 単語	207417	200098	12.97
		100 単語	221459	210989	13.85

注 MIPS 換算式

MIPS 値 = (処理に必要なサイクル数 ÷ 1 フレーム (16ms) 間のサイクル数) × DSP の MIPS 値

【最大認識単語数】

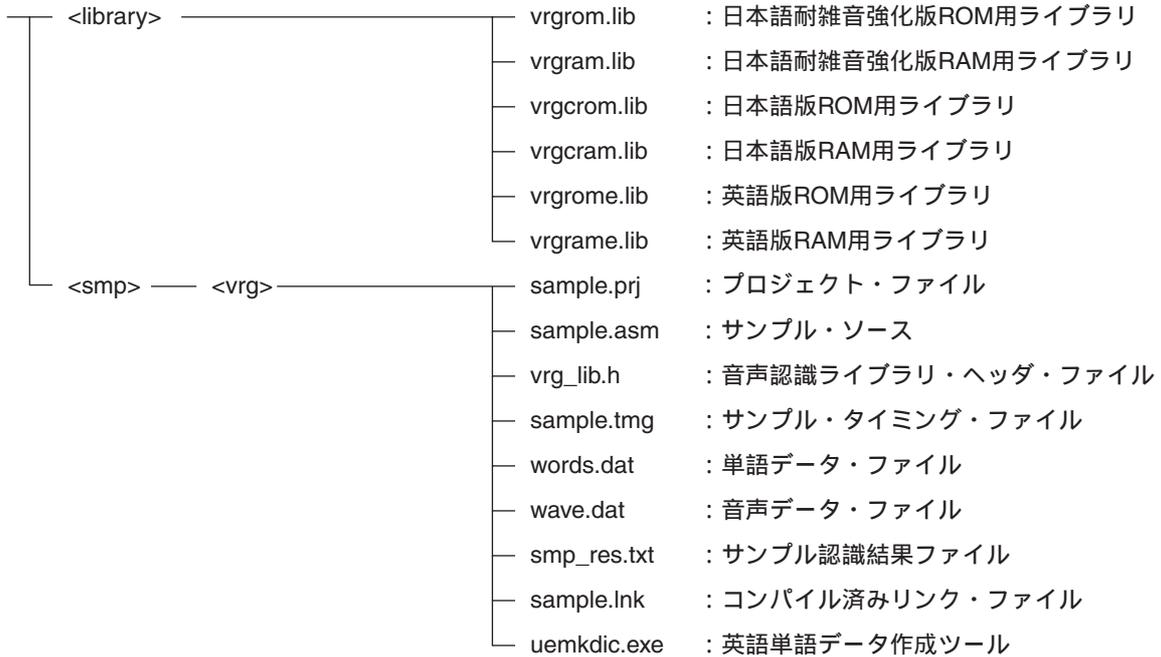
表 1-4 最大認識単語数

対象デバイス	日本語耐雑音強化版	日本語版	英語版
μ PD77110	240	260	-
μ PD77111, 77112	-	70	-
μ PD77113, 77113A, 77114	260	260	260
μ PD77115	-	100	-

備考 上記の値は，認識候補単語として日本語は1単語5文字，英語は1単語10音素を使用した場合です。

1.3.6 パッケージ内容

本ミドルウェアのディレクトリ構成を次に示します。



次に、各ディレクトリの概要を示します。

(1) <library>

各ライブラリ・ファイルを格納しています。

(2) <smp> - <vrg>

サンプル・プログラムのソース・ファイル，ヘッダ・ファイル，タイミング・ファイルを格納しています。サンプル・プログラムでは，vrgcram.lib のライブラリ・ファイルを使用します。サンプル・プログラムの使用法は，**第5章 システム例**を参照してください。

uemkdic.exe は英語版ライブラリ使用時に認識辞書に登録するための単語データを作成するツールです。詳しい説明は，**第4章 PC 版英語単語データ作成ツール**を参照してください。

第2章 ライブラリ仕様

2.1 ライブラリ概要

ライブラリ関数一覧を次に示します。

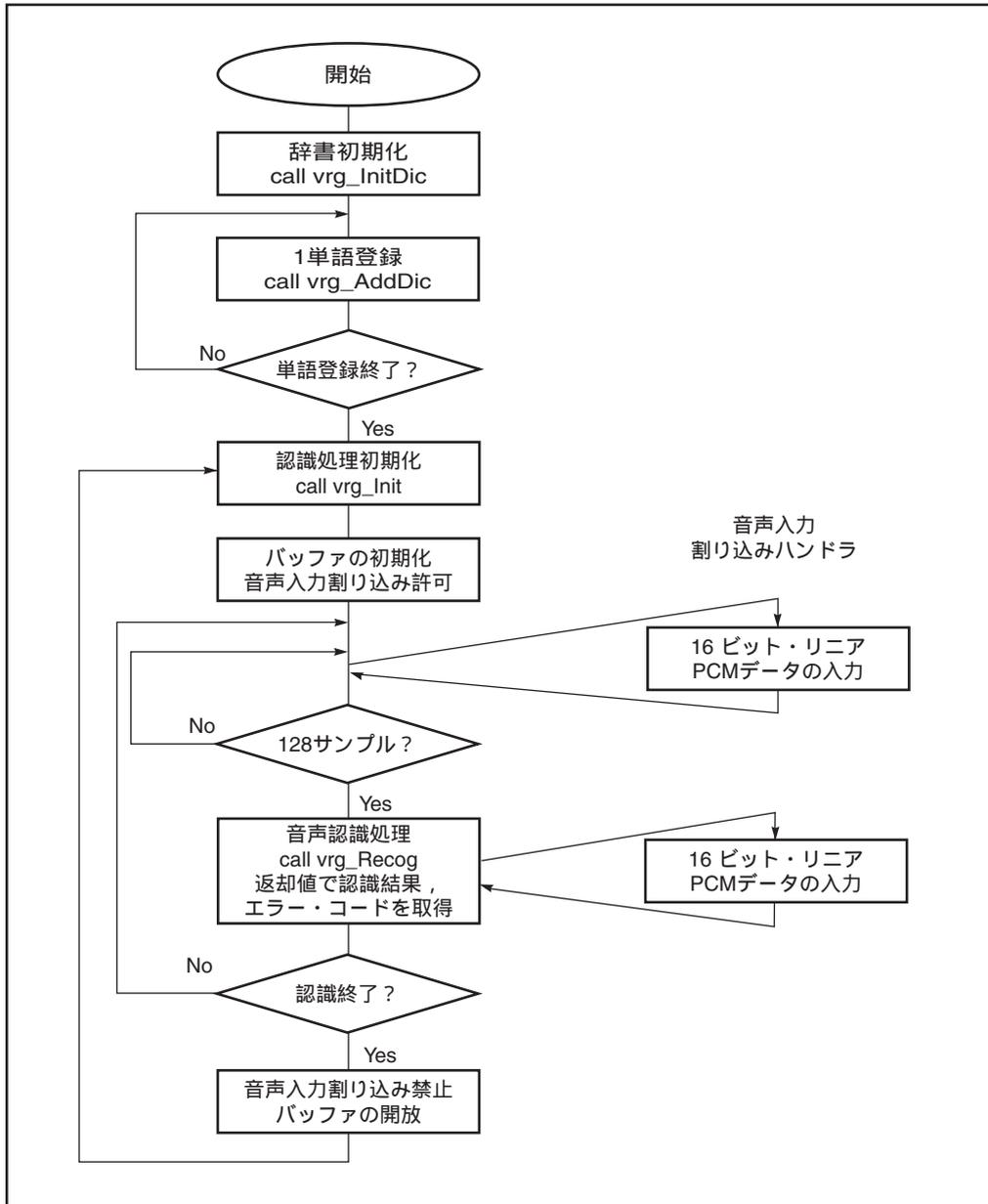
表 2-1 ライブラリ関数一覧

関数名	機能概要
vrg_InitDic	認識辞書の初期化を行います。
vrg_AddDic	認識辞書に 1 単語を追加します。
vrg_DelDic	認識辞書から 1 単語を削除します。
vrg_Init	認識処理の初期化を行います。
vrg_Recog	1 フレーム分の認識処理を行います。
vrg_GetVersion	本モデルウェアのバージョン情報を出力します。

2.2 アプリケーション・フロー

本モデルウェアを使用したシングル・タスク・アプリケーションの処理フロー例を図 2-1 に示します。

図 2-1 シングル・タスク・アプリケーション処理フロー例

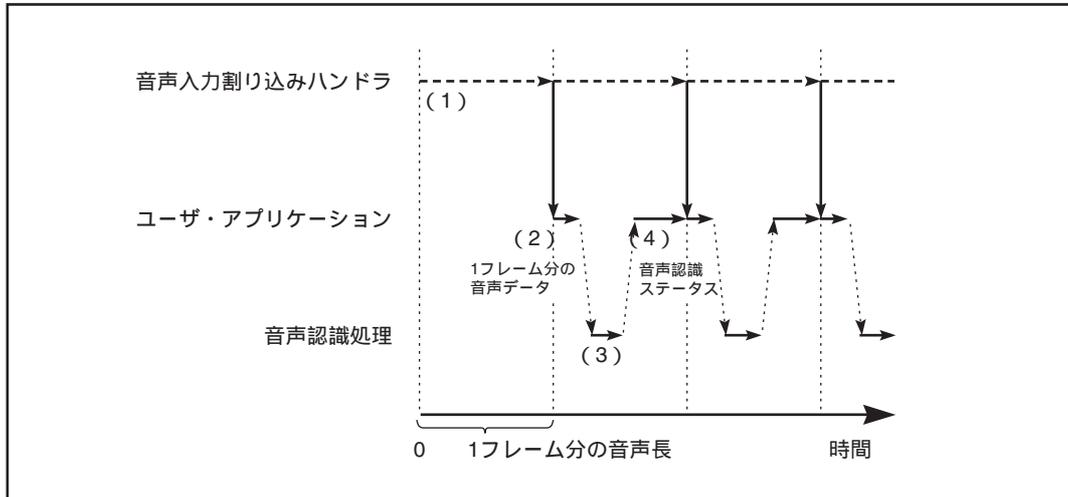


備考 音声データ入力処理部は、ターゲット・システムのハードウェアに依存しますので、ターゲット・システムにあわせて設計してください。

2.3 タイミング・ダイアグラム

次に、タイミング・ダイアグラムを示します。

図 2-2 タイミング・ダイアグラム



- (1) サンプリング周波数 8 kHz，符号付き 16 ビットの PCM データをバッファリング。
 - (2) 1 フレーム分 (128 サンプル) の PCM データがバッファにたまったら、vrg_Recog 関数をコール。
 - (3) 1 フレーム分のサンプル・データで音声認識処理を実行。
 - (4) 音声認識ステータス (音声検出中，認識終了など) をユーザ・アプリケーションへ返却。
- 余った時間は，必要に応じてユーザがアプリケーション処理に使用可。

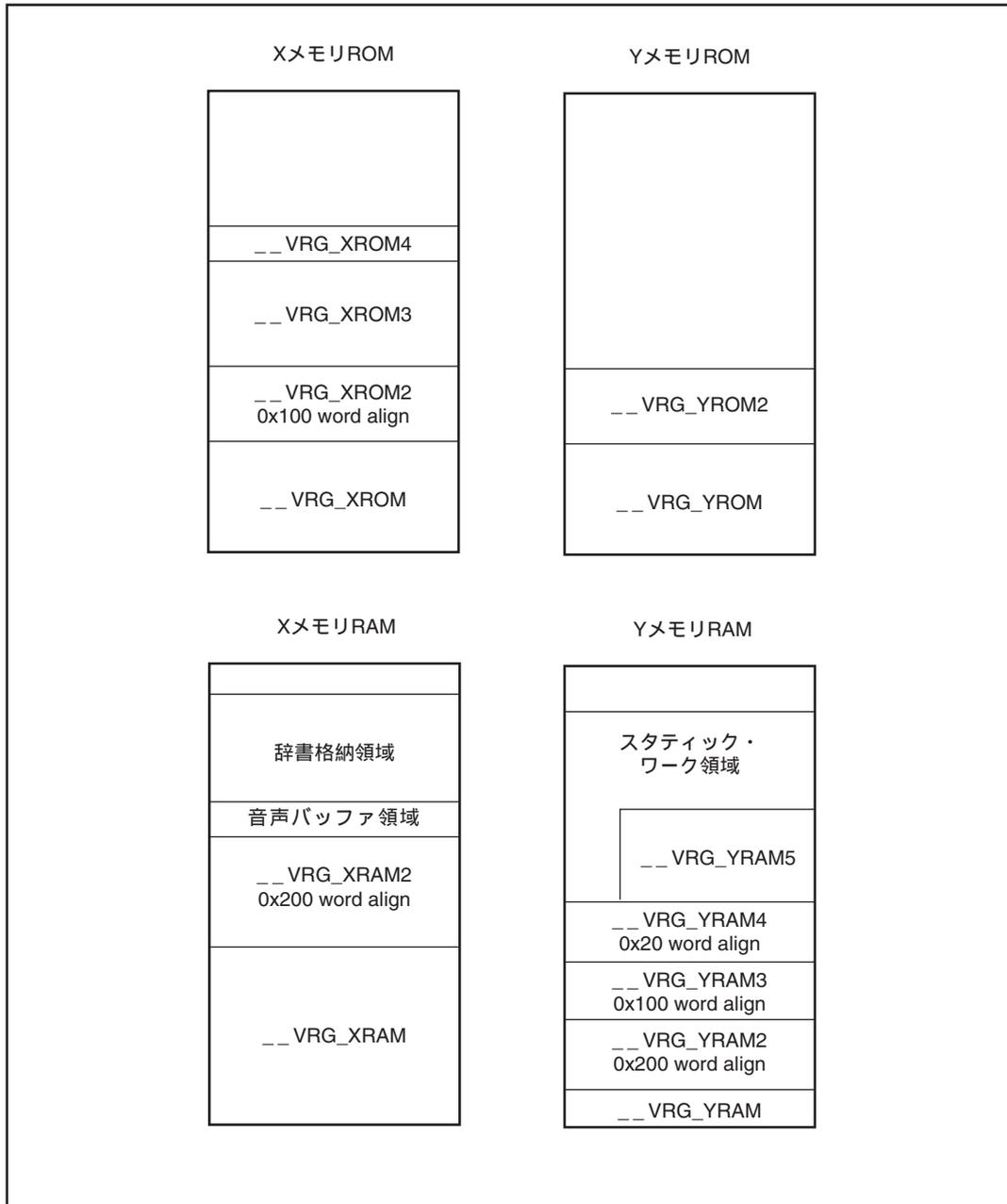
2.4 メモリ・マップ

本ミドルウェアを使用したアプリケーションのメモリ・マップ例を次に示します。

辞書格納領域とスタティック・ワーク領域は登録する認識単語数によって必要量が増減します。

ミドルウェアが使用する RAM 領域は，vrg_init 関数をコールしてから，1 発声の認識処理が終了するまでの間以外は，アプリケーションで自由に使用できます。

図 2-3 メモリ・マップ例



2.5 関数仕様

2.5.1 vrg_InitDic 関数

- 【分類】 辞書初期化関数
- 【関数名】 vrg_InitDic
- 【機能概要】 認識に使用する辞書の初期化を行います。
- 【形式】 call vrg_InitDic
- 【引き数】 DP0 辞書格納領域アドレス
R0L 辞書格納領域サイズ(ワード)
- 【戻り値】 処理結果: R0

表 2-2 vrg_InitDic 関数戻り値一覧

戻り値	内容
0	正常終了
-0x101 (0xFFFFFFFFE)	辞書格納領域サイズ・エラー 引き数 R0L に、最低限必要な辞書領域サイズである 9 ワードより小さな値が指定されました。

【コール・タイミング】 辞書に単語を登録する前に一度コールします。

【使用レジスタ】 R0, R1, R2, DP0, DP1

【ハードウェア・リソースメント】

最大スタック・レベル	1
最大ループ・スタック・レベル	1
最大リピート回数	0

【機能】 引き数で指定された辞書格納領域の初期化、各種パラメータの設定を行います。引き数の DP0 の値は関数終了時も保持されます。

2.5.2 vrg_AddDic 関数

【分類】	単語追加関数
【関数名】	vrg_AddDic
【機能概要】	認識に使用する辞書に1単語を追加します。
【形式】	call vrg_AddDic
【引き数】	DP0 辞書格納領域アドレス DP1 登録単語文字列アドレス
【戻り値】	処理結果：R0

表 2-3 vrg_AddDic 関数戻り値一覧

戻り値	内容
0を越える値	必要スタティック・ワーク・サイズ(正常終了時)
-0x201 (0xFFFFFFFFDFE)	未初期化エラー vrg_InitDic 関数を実行せずに、本関数がコールされました。
-0x202 (0xFFFFFFFFDFE)	辞書格納領域不足エラー vrg_InitDic 関数実行時に指定された辞書領域サイズに入りきらなくなりました。
-0x203 (0xFFFFFFFFDFD)	単語文字数オーバーエラー 引き数 DP1 で指定された単語文字列が日本語時 40 文字、英語時 80 ワードを越えています。
-0x204 (0xFFFFFFFFDFC)	単語文字コード・エラー 引き数 DP1 で指定された単語文字列中に不正なコードが含まれています。
-0x205 (0xFFFFFFFFDFB)	単語文字数ゼロ・エラー 引き数 DP1 で指定された単語文字列の長さがゼロです。

【コール・タイミング】 vrg_InitDic 関数コール後

【使用レジスタ】 R0, R1, R2, R3, R4, R5, R7, DP0, DP1, DP2, DP4, DP5, DP6, DP7

【ハードウェア・リソースメント】

最大スタック・レベル	6
最大ループ・スタック・レベル	1
最大リピート回数	0

【機能】

(a) 機能

本モデルウェアは、かな文字列を直接認識に使用することはできません。vrg_AddDic 関数を使用して専用バイナリ形式に変換したものを使用します。この専用バイナリ形式に変換したものを「辞書」と呼び、「専用バイナリ形式に変換する」ことを「辞書作成」と呼びます。英語版では、PC で動作する英語単語データ作成ツールを用いて専用バイナリ形式への変換を行ないます。英語単語データ作成ツールについては、第4章 PC 版英語単語データ作成ツールを参照してください。

日本語版では、引き数 DP1 によって指定されたアドレスを先頭とする登録単語文字列から辞書データを作成し、引き数 DP0 で指定された辞書格納領域中に出しします。英語版では、PC 上で英語単語データ作成ツールを用いて単語データの作成を行ない、その単語データの先頭アドレスを引き数 DP1

に指定してください。引き数 DP0 で指定された辞書格納領域中に出力します。

DP0 には、vrg_InitDic 関数コール時に指定したものと同一アドレスを指定してください。また、単語が辞書に登録されると同時に単語 ID が割り振られます。認識結果は、この単語 ID で返却されます。単語 ID については、2.5.2 (d) 単語 ID を参照してください。

(b) 戻り値

正常終了した場合、引き数で指定した辞書格納領域を使用して音声認識を行う際に必要なスタティック・ワーク領域サイズを戻り値として、R0 に返却します。音声認識を行うには、この必要スタティック・ワーク・サイズ以上のスタティック・ワーク領域が必要です。十分なスタティック・ワーク領域を確保していることを確認してください。

エラーが返却された場合、辞書格納領域の辞書データは vrg_AddDic 関数がコールされる前の状態に保存されています。

また、メモリの内容によっては、vrg_InitDic 関数で初期化せずに、Vrg_AddDic 関数をコールした場合でも未初期化エラーが返却されない場合があります。

(c) 入力フォーマット(日本語版)

日本語の入力フォーマットは、1 単語分の単語文字列の後に、終端として 0x0000 を付加したものです。文字列格納イメージは図 2-4 に示すとおりです。登録単語文字列は、シフト JIS コードの全角ひらがな、全角カタカナおよび長音記号で指定してください。使用可能コードは表 2-4 に、使用可能文字は表 2-5 に示すとおりです。

図 2-4 登録単語文字列格納イメージ (5 文字の場合)

1文字目のコード
2文字目のコード
3文字目のコード
4文字目のコード
5文字目のコード
終端 (0x0000)

表 2-4 登録単語文字列使用可能コード一覧

(i) 長音記号「 ー 」(0x815B)

(マイナス記号「 - 」(0x917C)とは異なります)

(ii) 最上位 8 ビットが 0x82 のコード

		下位 4 ビット															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
上位 4 ビット	9	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	ぁ
	A	あ	い	い	う	う	え	え	お	お	か	が	き	ぎ	く	ぐ	け
	B	げ	こ	こ	さ	ざ	し	じ	す	ず	せ	ぜ	そ	ぞ	た	だ	ち
	C	ぢ	っ	っ	づ	て	で	と	ど	な	に	ぬ	ね	の	は	ば	ぱ
	D	ひ	び	び	ふ	ぶ	ぶ	へ	べ	べ	ほ	ぼ	ぼ	ま	み	む	め
	E	も	ゃ	ゃ	ゅ	ゅ	よ	よ	ら	り	る	れ	ろ	わ	わ	ゐ	ゑ
	F	を	ん	/	/	/	/	/	/	/	/	/	/	/	/	/	/

(iii) 最上位 8 ビットが 0x83 のコード

		下位 4 ビット															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
上位 4 ビット	4	ァ	ア	ィ	イ	ウ	ウ	ェ	エ	ォ	オ	カ	ガ	キ	ギ	ク	グ
	5	ケ	ゲ	コ	ゴ	サ	ザ	シ	ジ	ス	ズ	セ	ゼ	ソ	ゾ	タ	ダ
	6	チ	ヂ	ツ	ツ	ッ	テ	デ	ト	ド	ナ	ニ	ヌ	ネ	ノ	ハ	バ
	7	パ	ピ	ピ	ピ	フ	ブ	ブ	ヘ	ベ	ベ	ホ	ボ	ポ	マ	ミ	/
	8	ム	メ	モ	ャ	ヤ	ユ	ユ	ヨ	ヨ	ラ	リ	ル	レ	ロ	ワ	ワ
	9	ヰ	ヱ	ヲ	ン	ヴ	カ	ケ	/	/	/	/	/	/	/	/	/

表 2-5 使用可能文字一覧表

使用方法およびその可否	文 字																		
使用可能文字	あ い う え お か き く け こ さ し す せ そ た ち つ て と な に ぬ ね の は ひ ふ へ ほ ま み む め も や ゆ よ ら り る れ ろ が ぎ く げ ご ざ じ ず ぜ ぞ だ ぢ づ で ど ば び ぶ べ ぼ ば び ぶ べ ぼ ー わ ゐ ゑ を アイウエオカキクケコサシスセソ タチツテトナニヌネノハヒフヘホ マミムメモヤユヨラリルレロ ガギグゲゴザジズゼゾダヂヅデド バビブベボバビブベボヴワヰヱヲ																		
単語の先頭では使用不可	ん っ ン ッ																		
特殊な文字との組み合わせでのみ使用可	<table border="1"> <thead> <tr> <th>文字</th> <th>下記のいずれかの文字の後でのみ使用可能</th> </tr> </thead> <tbody> <tr> <td>ぁ ア</td> <td>ふ フ ヴ</td> </tr> <tr> <td>い イ</td> <td>う て で ふ ウ テ デ フ ヴ</td> </tr> <tr> <td>う ウ</td> <td>と ど ふ ト ド フ ヴ</td> </tr> <tr> <td>え エ</td> <td>う し じ ち ぢ つ ぶ ウ シ ジ チ ヂ ツ フ ヴ</td> </tr> <tr> <td>ぉ オ</td> <td>う ふ ウ フ ヴ</td> </tr> <tr> <td>ゃ よ</td> <td>き し ち に ひ み り ぎ じ ぢ び び</td> </tr> <tr> <td>ゃ ヨ</td> <td>キ シ チ ニ ヒ ミ リ ギ ジ チ ビ ビ ヴ</td> </tr> <tr> <td>ゅ ユ</td> <td>き し ち て に ひ み り ぎ じ ぢ で び び キ シ チ テ ニ ヒ ミ リ ギ ジ チ デ ビ ビ ヴ</td> </tr> </tbody> </table>	文字	下記のいずれかの文字の後でのみ使用可能	ぁ ア	ふ フ ヴ	い イ	う て で ふ ウ テ デ フ ヴ	う ウ	と ど ふ ト ド フ ヴ	え エ	う し じ ち ぢ つ ぶ ウ シ ジ チ ヂ ツ フ ヴ	ぉ オ	う ふ ウ フ ヴ	ゃ よ	き し ち に ひ み り ぎ じ ぢ び び	ゃ ヨ	キ シ チ ニ ヒ ミ リ ギ ジ チ ビ ビ ヴ	ゅ ユ	き し ち て に ひ み り ぎ じ ぢ で び び キ シ チ テ ニ ヒ ミ リ ギ ジ チ デ ビ ビ ヴ
文字	下記のいずれかの文字の後でのみ使用可能																		
ぁ ア	ふ フ ヴ																		
い イ	う て で ふ ウ テ デ フ ヴ																		
う ウ	と ど ふ ト ド フ ヴ																		
え エ	う し じ ち ぢ つ ぶ ウ シ ジ チ ヂ ツ フ ヴ																		
ぉ オ	う ふ ウ フ ヴ																		
ゃ よ	き し ち に ひ み り ぎ じ ぢ び び																		
ゃ ヨ	キ シ チ ニ ヒ ミ リ ギ ジ チ ビ ビ ヴ																		
ゅ ユ	き し ち て に ひ み り ぎ じ ぢ で び び キ シ チ テ ニ ヒ ミ リ ギ ジ チ デ ビ ビ ヴ																		
「っ」「ッ」のあとで使用不可	あ い う え お な に ぬ ね の ま み む め も や ゆ よ ら り る れ ろ わ ゐ ゑ を ん うい うえ うお にや にゆ によ みや みゆ みよ - りゃ りゆ りよ あ い う え お や ゆ よ っ わ アイウエオナニヌネノマミムメモ ヤユヨラリルレロワヰヱヲ ウイ ウェ ウオ ニヤ ニユ ニョ ミヤ ミユ ミヨ リヤ リユ リヨ アイウエオヤユヨ ッ ワ																		
末尾で使用不可	っ ッ																		
「 - 」の前では使用不可	ん っ ン ッ																		
その他、使用不可の組み合わせ	んん んン ンん ンン																		

(d) 単語 ID

単語 ID は、vrg_AddDic 関数で登録された単語の順に先頭から、0,1,2,...と割り振られています。

たとえば、「さっぽろ」「せんだい」「ちば」「かわさき」「よこはま」「なごや」「きょうと」「おおさか」「こうべ」「ひろしま」「きたきゅうしゅう」「ふくおか」の順に 12 単語が登録されている場合は、表 2-6 のようになります。

表 2-6 単語 ID 列

登録単語	単語 ID
さっぽろ	0
せんだい	1
ちば	2
かわさき	3
よこはま	4
なごや	5
きょうと	6
おおさか	7
こうべ	8
ひろしま	9
きたきゅうしゅう	10
ふくおか	11

(e) 辞書フォーマット

辞書格納領域に出力される辞書データは、次のような構造になっています。

登録済み単語数合計や、実際に使われている辞書サイズが知りたい場合には、辞書格納領域を参照してください。必要スタティック・ワーク・サイズは、最後に vrg_AddDic 関数をコールして正常終了した際に返却された値と同じです。

図 2-5 辞書格納領域 (1ワード単位)

登録済み単語数合計
使用済み辞書領域サイズ
必要スタティック・ワーク・サイズ
辞書データ本体 : :
終端 (0x0000)

2.5.3 vrg_DelDic 関数

- 【分類】 単語削除関数
- 【関数名】 vrg_DelDic
- 【機能概要】 認識に使用する辞書から 1 単語を削除します。
- 【形式】 call vrg_DelDic
- 【引数】 DP0 辞書格納領域アドレス
R0L 削除単語の単語 ID
- 【戻り値】 処理結果：R0

表 2-7 vrg_DelDic 関数戻り値一覧

戻り値	内容
0	正常終了
-0x301 (0xFFFFFFFFFCFF)	未初期化エラー vrg_InitDic 関数を呼ばずに、本関数がコールされました。
-0x302 (0xFFFFFFFFFCFE)	指定単語未登録エラー 引き数 R0L で指定された単語 ID を持つ単語は存在しませんでした。

- 【コール・タイミング】 vrg_AddDic 関数コール後
- 【使用レジスタ】 R0, R1, R2, R3, R4, R5, DP0, DP1, DP2, DP4
- 【ハードウェア・リソースメント】
- | | |
|----------------|---|
| 最大スタック・レベル | 1 |
| 最大ループ・スタック・レベル | 1 |
| 最大リピート回数 | 0 |

- 【機能】 辞書から 1 単語を削除します。引き数に削除したい単語の単語 ID を指定します。該当単語が存在した場合、該当単語を削除し、該当単語以降に登録されていた単語の単語 ID が 1 ずつ減らされます。

たとえば、表 2-6の状態では、vrg_DelDic 関数をコールして単語 ID5 の「なごや」を削除すると、「きょうと」の単語 ID が 5 に、「おおさか」の単語 ID が 6 になります。同様にそれ以降の単語も 1 ずつ単語 ID が減らされます。

2.5.4 vrg_Init 関数

- 【分類】 認識初期化関数
- 【関数名】 vrg_Init
- 【機能概要】 認識処理の初期化を行います。
- 【形式】 call vrg_Init
- 【引き数】 DP0+0 辞書格納領域アドレス (XRAM)
 DP0+1 スタティック・ワーク領域アドレス (YRAM)
 DP0+2 スタティック・ワーク領域サイズ

```

例 dp0 = LibEntry;
    r0l = Dictionary;
    *dp0++ = r0l;           // Dictionary Address (XRAM)
    r0l = StaticWork;
    *dp0++ = r0l;         // StaticWork Address (YRAM)
    r0l = StaticWorkSize;
    *dp0++ = r0l;         // StaticWork Size
    dp0 = LibEntry;
  
```

- 【戻り値】 処理結果：R0

表 2-8 vrg_Init 関数戻り値一覧

戻り値	内容
0	正常終了
-0x401 (0xFFFFFFFFBFF)	辞書未初期化エラー vrg_InitDic 関数で辞書を初期化せずに、本関数がコールされました。
-0x402 (0xFFFFFFFFBFE)	単語未登録エラー vrg_AddDic 関数で辞書に単語が登録せずに、本関数がコールされました。
-0x403 (0xFFFFFFFFBFD)	スタティック・ワーク・サイズ・ゼロ以下エラー 引き数で指定されたスタティック・ワーク領域サイズが0以下の値です。
-0x404 (0xFFFFFFFFBFC)	スタティック・ワーク・サイズ不足エラー 引き数で指定されたスタティック・ワーク領域サイズが、辞書の内容から求められる最低限必要なスタティック・ワーク領域サイズよりも少ないです。

- 【コール・タイミング】 各発声ごとに、認識処理に先立ってコールします。

- 【使用レジスタ】 R0, R1, R2, R7, DP0, DP4, DP6

【ハードウェア・リソースメント】

最大スタック・レベル	5
最大ループ・スタック・レベル	1
最大リピート回数	0

- 【機能】 認識処理が使用するメモリ領域の初期化および各種パラメータの設定を行います。引き数で指定された辞書，スタティック・ワーク領域を認識処理で使用します。辞書格納領域とスタティック・ワーク領域のメモリ配置については，**図 2-3 メモリ・マップ例**を参照してください。
- 引き数で指定された辞書格納領域が，辞書初期化関数 `vrg_InitDic` で初期化されていない場合，辞書未初期化エラーが返却されます。また，単語がまったく登録されていない場合は，単語未登録エラーが返却されます。
- 引き数で指定されたスタティック・ワーク領域サイズが 0 以下の場合，スタティック・ワーク・サイズ・ゼロ以下エラーが返却されます。スタティック・ワーク領域の必要サイズは辞書の内容に依存します。引き数で指定されたスタティック・ワーク領域サイズが，必要量に満たない場合は，スタティック・ワーク・サイズ不足エラーが返却されます。

2.5.5 vrg_Recog 関数

【分類】	認識処理関数
【関数名】	vrg_Recog
【機能概要】	1 フレーム分の音声認識処理を行います。
【形式】	call vrg_Recog
【引き数】	DP0 1 フレーム分の音声 (128 サンプル×符号付き 16 ビット) バッファ・アドレス
【戻り値】	処理結果 : R0 第 1 位認識候補単語 ID : R1

表 2-9 vrg_Recog 関数戻り値 R0 一覧

戻り値	内 容
0	認識終了
1	音声未検出
2	音声検出中
-0x501 (0xFFFFFFFFAFF)	未初期化エラー vrg_Init 関数で初期化せずに、本関数がコールされました。
-0x502 (0xFFFFFFFFAFE)	コール回数オーバーエラー 32768 回以上、本関数がコールされました。vrg_Init 関数で再度初期化してください。
-0x503 (0xFFFFFFFFAFD)	致命的エラー 予期しないエラーが発生しました。メモリ内容が不正に書き換えられている可能性があります。

【コール・タイミング】 1 フレーム分の音声データが準備されるごとにコール

【使用レジスタ】 R0, R1, R2, R3, R4, R5, R6, R7, DP0, DP1, DP2, DP3, DP4, DP5, DP6, DN0, DN1, DN2, DN3, DN4, DMX, DMY

【ハードウェア・リソースメント】

最大スタック・レベル	6
最大ループ・スタック・レベル	3
最大リピート回数	31

【機能】

(a) 動作

引き数で指定された音声データに対して、1 フレーム分の音声認識処理を行います。vrg_Init 関数コール時に指定された辞書およびスタティック・ワーク領域を使用します。

1 単語発声分の認識処理が終了し、認識結果が得られた場合は、R0 に 0 が戻り値として返却され、R1 には第 1 位認識候補単語 ID が返却されます。認識が終了していない場合には、音声未検出状態であれば R0 に 1 が、音声検出状態であれば、R0 に 2 が返却され、R1 に 0 が返却されます。

vrg_Recog 関数は、1 発声の認識について最大 32767 回までしかコールすることはできません。それを越えてコールした場合、コール回数オーバーエラーが返却されます。再度、認識を開始するには、vrg_Init 関数をコールしてください。

(b) 詳細認識結果

認識結果の詳細は、シンボル `vrg_result` 以降に図 2-6のような構造で格納されます。この領域は、`vrg_init` 関数コール時に初期化され、認識終了時に値がセットされます。

単語 ID、距離値は-1~0x7FFF の範囲をとります。距離値とは、入力された音声データと認識候補単語パターンとの離れ具合を表します。距離値が小さいほど信頼性が高くなります。

辞書登録単語数が 4 単語以下の状態で音声認識を実行した場合など、認識候補として該当する単語がない場合は、認識候補単語 ID、距離値はともに-1 となります。また、認識候補単語距離値がすべて同じになった場合にも、単語 ID、距離値ともに-1 の値をとります。

図 2-6 `vrg_result` (1ワード単位)

第1位認識候補単語ID
第2位認識候補単語ID
第3位認識候補単語ID
第4位認識候補単語ID
第5位認識候補単語ID
第1位認識候補単語距離値
第2位認識候補単語距離値
第3位認識候補単語距離値
第4位認識候補単語距離値
第5位認識候補単語距離値
最大音声レベル
背景雑音レベル
S/Nレベル
認識発生時間 (msec)

2.5.6 vrg_GetVersion 関数

- 【分 類】 バージョン情報取得関数
- 【関 数 名】 vrg_GetVersion
- 【機能概要】 本ミドルウェアのバージョン情報を出力します。
- 【形 式】 call vrg_GetVersion
- 【引 き 数】 なし
- 【戻 り 値】 R0H メージャ・バージョン番号
R0L マイナ・バージョン番号
- 【使用レジスタ】 R0
- 【ハードウェア・リソースメント】
- | | |
|----------------|---|
| 最大スタック・レベル | 1 |
| 最大ループ・スタック・レベル | 0 |
| 最大リピート回数 | 0 |
- 【機 能】 本ミドルウェアのバージョン番号を 32 ビットの値で出力します
- 例 R0 = 0x00'0x0001'0x0100 の場合
バージョン : V1.01

第3章 インストール

3.1 インストール手順

本ミドルウェアの供給媒体は、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の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. 6 パッケージ内容**を参照してください。

```
a:¥>dir c:¥DSPTools<CR>
```

3.2 サンプル・プログラム作成手順

提供媒体のディレクトリ (smp) 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 認識処理サンプル・ソース (sample.asm)**を参照)。

サンプル・プログラムは、ハイスピード・シミュレータ HSM77016 上で、後述するタイミング・ファイルを使用することで、単語や音声データの入出力をシミュレーション可能にします。タイミング・ファイルについては、**5.2 操作方法**を参照してください。

次に音声認識ミドルウェアのサンプル・プログラムのビルド方法について例を示します。

- (1) WB77016 (ワークベンチ) を起動します。
- (2) sample.prj プロジェクトを開きます。
例 「Project → Open Project」で sample.prj を指定します。
- (3) モデル・ファイルとして、uPD77115.model を指定します。
例 「Options → Processor Model」で uPD77115.model を指定します。
- (4) 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 → Build All」を選択すると、sample.lnk ファイルが生成されます。

サンプル・プログラムの実行方法は、**第5章 システム例**を参照してください。

3.3 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を組み合わせで使用するときは、重複しないようにしてください。

表 3-1 シンボル名

分 類	シンボル名
関数, 変数名	vrg_xxxx
マクロ, 定数名	VRG_XXXX
セクション名	__VRG_XXXX (先頭のアンドスコアは2つ)

第4章 PC 版英語単語データ作成ツール

4.1 動作環境

Windows95/98 上の DOS プロンプト

4.2 単語データ作成方法

コマンド・ラインから次のコマンドを入力して実行します。

```
c:¥> uemkdic [-r] input <CR>
```

表 4-1 PC 版英語単語データ作成ツール引き数

引き数	機能
-r	認識単位 ID 列の重複をチェック
input	ソース単語ファイル名

正常終了[※]時は 0 を、エラーを検出した場合は負の値を返します。

注 エラーを検出しなかった場合は、ワーニングがあっても正常終了とみなします。

4.3 入力ファイル

単語データ作成ツールへ入力するソース単語ファイルは、ASCII コードで記述したテキスト・ファイルです。

ソース単語ファイルは、4.3.1 音素記号一覧の音素記号を用いて、各単語の発音を記述します。

各音素記号は 1 つ以上の区切り文字で区切られ、区切り文字には、タブ (09H) またはスペース (20H) を使用します (行頭に区切り文字を入れるかどうかは任意です)。

1 単語あたりの最大音素記号数は 30、1 行の最大バイト数は 256 バイト (改行文字も含まれます) なので、この範囲内で記述してください。

1 単語につき 1 行を使用します。ただし、空白文字のみの行があってもかまいません。

改行文字は CR / LF とします (LF のみの場合の動作は保証しません)。

‘;’ (セミコロン) から行末まではコメントとみなします。

例を次に示します。

```

ax l ae s cl k ax           ; Alaska
ae r ih z ow n ax          ; Arizona
ar cl k ax n s ao          ; Arkansas
    
```

⏟

発音 (必須)

⏟

コメントとして単語の表記を記入 (必須ではありません)

4.3.1 音素記号一覧

次に、ソース単語ファイルに使用する音素記号の一覧を示します。

(1) 母音：22種類

ih, iy, eh, ae, ah, o, aa, ao, uh, uw, rr, ax, ey, ay, oy, aw, ow, ir, er, ar, or, ur

(2) 子音：24種類

p, b, t, d, k, g, ch, jh, f, v, th, dh, s, z, sh, zh, hx, y, w, r, m, n, nx, l

(3) その他：2種類

cl, *cl

4.3.2 破裂音の記法

破裂音 (p, t, k, ch, b, d, g, jh) を記述する場合、発話始端ではクロージャ・モデル (cl, *cl) を挿入せず、発話始端以外の位置ではクロージャ・モデルを破裂音の前に必ず挿入しなければなりません。破裂音の記述法の一覧を次に示します。

音素記号	p	t	k	ch	b	d	g	jh
発話始端	p	t	k	ch	b	d	g	jh
発話始端以外の位置	cl p	cl t	cl k	cl ch	*cl b	*cl d	*cl g	*cl jh

4.3.3 音素記号

単語データ作成ツールは、“Merriam Webster’s Collegiate Dictionary”で使用されている発音記号に、ASCIIコードを割り当てています。

単語作成にあたっては、次のコードを参照してください。

表 4-2 母音 (Vowels)

Merriam Webster	IPA	ASCII	Word(Example)
ē	i:	iy	beat
i	i	ih	bit
ir	iə r	ir	year
e	e	eh	bet
a	æ	ae	bat
ā	ei	ey	bait
er	ɛər	er	where
ə	ə	ax	the
‘ə	ʌ	ah	butt
ər	ə:r	rr	bird
ü	u:	uw	boot
ù	u	uh	book
úr	uər	ur	poor
ó	ɔ:	ao	bought
ä	ɑ	o	cot
á	ɑ:	aa	father
ī	ai	ay	bite
ói	ɔi	oy	boy
áu	au	aw	out
ō	ou	ow	boat
är	ɑ:r	arl	car
ór	ɔ:r	or	door

表 4-3 子音 (Consonants)

Merriam Webster	IPA	ASCII	Word(Example)
h	h	hx	hay
l	l	l	led
r	r	r	red
y	j	y	yet
w	w	w	wet
m	m	m	mom
n	n	n	non
ŋ	ŋ	nx	sing
j	dʒ	jh	judge
b	b	b	bob
d	d	d	dad
g	g	g	gag
p	p	p	pop
t	t	t	tot
k	k	k	cat
ch	tʃ	ch	church
th	ð	dh	they
z	z	z	zoo
zh	ʒ	zh	measure
v	v	v	very
f	f	f	fief
th	θ	th	thief
s	s	s	sister
sh	ʃ	sh	shoe

	ASCII
Silence	#
Closure(Unvoiced)	cl
Closure(Voiced)	*cl

4.3.4 Merriam Webster から ASCII 発音記号変換ルールについて（補足）

表 4-4 は、音素記号の説明に際して、Merriam Webster 辞書と対応付けて説明しています。

しかし、ah にあたる記号も ax にあたる記号も Merriam Webster では、「 θ 」であるために、発音記号の混同が起きる可能性があります。

補足説明として、ストレス位置に関するルールを表 4-5 に示します。

英単語発音には、primary stress と secondary stress の 2 種類が用いられることもあります。たとえば、'communicate' という単語では、'mmu' の母音に primary stress、'cate' の母音に secondary stress、があります。

母音が「 θ 」であるような音節に上記の primary あるいは secondary のストレス印がついていたら、ah を用いるか、または ax を用いるようにしてください。

表 4-4 IPA と Merriam Webster の発音記号対応

ASCII 音素記号	単語例	IPA	Merriam Webster
ah	butt	\wedge	θ
ax	the	θ	θ

表 4-5 ストレス位置

ASCII 音素記号	単語例	Merriam Webster
ah	butt	' θ (ストレスあり)
ax	the	θ (ストレスなし)

結論として、アクセントがある場合には同じ発音記号でも、DSP 英語認識用単語は、異なる ASCII 発音記号を割り当てる必要があることになります。

4.4 出力ファイル

処理結果として辞書ファイルを出力します。

辞書ファイルは、入力ファイル名の拡張子を“.ued”に置換したバイナリ・ファイルとして出力されます。同名のファイルがすでに存在していた場合は、既存のファイルに上書きされます。エラーが発生した場合には、出力されません。

辞書ファイルには、単語ごとのデータが単語数分、NULL を区切りとして書き込まれています。データはすべて16ビット(リトル・エンディアン)単位です。

単語ごとのデータ	NULL	単語数分繰り返し・・・
----------	------	-------------

4.5 エラー・コード

辞書作成処理でエラーが発生した場合は、次のエラー情報が標準出力に表示されます（カッコ内の情報は、該当するものがある場合だけ表示します）

エラー番号が一桁の場合は、致命的エラーです。致命的エラーが発生しないかぎりには、入力ファイルに含まれるすべてのエラーを出力します。エラー番号とエラー内容、およびその処置を表 4-6 に示します。

Error エラー番号（行番号：該当行の記述内容）

表 4-6 エラー・コード一覧

エラー番号	エラー内容	処 置
1	メモリ・オーバ ^{※1}	ソース辞書ファイルの行数を少なくするか、実行中のほかのアプリケーションを終了してください。
2	ファイル・オープン・エラー	本プログラムがファイルを出力できる ^{※2} ように、ソース辞書ファイルのファイル名を正しく入力してください。
3	システム・エラー	PCを再起動したあと、再試行してください ^{※3} 。
4	ファイル・リード・エラー	ソース辞書ファイルのあるハードディスクに異常がないかをチェックして ^{※4} 、異常がある場合は修復してください。
5	ファイル・ライト・エラー	本プログラムがファイルを出力するハードディスクに異常がないかをチェックして ^{※4} 、異常がある場合は修復してください。ファイルを出力するための空き容量が不足している場合は、空き容量を増やしてください。
101	文字数オーバ	該当行が 256 バイト以内になるように、文字数を減らしてください。
102	音素記号数オーバ	ソース辞書ファイルの該当行を削除してください。1 単語の音素記号数が、本プログラムのスペックを越えているためにコンパイルできません。
103	破裂音記述エラー	破裂音を正しく記述してください。
104	音素記号記述エラー	音素記号を正しく記述してください。
105	音素記号もれエラー	ソース辞書ファイルに音素記号を記述してください。

- 注 1. コンパイルに必要なメモリ容量が PC に不足しているために発生するエラーです。DSP 上で音声認識を実行する際にメモリ不足になるという意味ではありません。
2. 本プログラムが出力しようとするファイルと同じ名前のファイルを、ほかのアプリケーションで開いている場合は、そのファイルを閉じてください。ファイルの属性が読み取り専用になっている場合は、その属性を解除するなどの処置をしてください。
3. 標準出力をファイルにリダイレクトして実行した場合は、リダイレクト先のハードディスクでエラーが発生した可能性があるため、ファイル・ライト・エラーと同様の処置も合わせて行ってください。
4. ハードディスクのエラー・チェック・プログラムを実行するなどの処置を行ってください。

第5章 システム例

5.1 タイミング・ファイルを使用したシミュレーション環境

シミュレータ上でタイミング・ファイルを使用し、音声認識を行なう例を次に示します。単語データと音声データをを入力し、1フレーム単位で音声認識処理を行ない、認識終了すると結果を出力します。

5.2 操作方法

- (1) HSM77016 (ハイスピード・シミュレータ) を起動します。
- (2) モデル・ファイルとして、uPD77115.model を指定します。
例 「Tools → Simulation Model」で uPD77115.model を指定します。
- (3) 3.2 サンプル・プログラム作成手順で作成した sample.lnk を開きます。
例 「file → open」で sample.lnk を指定します。
- (4) 次にタイミング・ファイル sample.tmg を開きます。
例 「file → open」で sample.tmg を指定します。
- (5) wait 設定を行います。
例 「Window → Periphery Register」で開く設定ウィンドウで、DWTR, IWTR の各レジスタに設定します。
- (6) Run で実行します。

5.3 動作説明

サンプル・タイミング・ファイルでは、次の手順で音声認識を行います。

- (1) 辞書ファイル (words.dat) を読み込み、HOST・インタフェースを使って入力します。具体的な記述例は
5.3.1 単語データの入力 (16 ビット・データ) を参照してください。
ここで使用する辞書ファイルは、各単語をシフト JIS 全角かな文字で 1 行に 1 文字ずつ記述したテキスト・ファイルです。単語の終端として 0x0 を、辞書の終端として 0xFFFF を記述します。
- (2) 音声認識開始の合図として、DSP 側から HOST・インタフェースに 0 が出力されるので、それを待ちます。

- (3) 音声データ・ファイル (wave.dat) を読み込み、シリアル・インタフェースを使って入力します。具体的な記述例は**5.3.2 音声データの入力 (16 ビット・データ)**を参照してください。1 フレーム分の音声データが入力されるごとに、DSP 側では平行して認識処理が実行されます。

ここで使用する音声データ・ファイルは、音声データの各サンプル値を 1 行に 1 ワードずつ記述したテキスト・ファイルです。

- (4) 認識結果をホスト I/F を使って出力し、認識結果ファイル (result.txt) に書き出します。具体的な記述例は**5.3.3 認識結果の出力**を参照してください。

ここで出力される認識結果ファイルは、**2.5.5 (b) 詳細認識結果**のシンボル vrg_result 以降のメモリ内容を 10 進数で 1 ワードずつそのまま出力したものです。認識結果例 (smp_res.txt) を見ると、1 行目に 0 が書かれています。このことから、音声データ (wave.dat) は、単語 ID0 の単語として認識されたことがわかります。音声データの内容は「ぜろ」であり、単語 ID0 の単語内容も「ぜろ」なので、正しく認識されたことになります。

5.3.1 単語データの入力 (16 ビット・データ)

音声認識辞書に登録する単語文字列を、ホスト・インタフェースによって入力します。次に記述例を示します。

・準備

```
open input "words.dat" ; 入力ファイル指定
input format hex ; 入力ファイルのフォーマット指定 (16進数)
```

・入力処理 (16 ビット・データ)

```
set pin hcs = 1 ; チップ・セレクトをインアクティブにする
set pin hwr = 1 ; ホスト・ライト・ストロブモインアクティブにする
set pin hrd = 1 ; ホスト・リード・ストロブモインアクティブにする
do
  wait cond pin hwe == 0 ; ホストがHTD(in)に書き込み可能になるまでウエイトする
  wait cond pin hcs == 1 ; チップ・セレクトがインアクティブになるまでウエイトする
  set pin hcs = 0 ; チップ・セレクトをアクティブにして、入力を開始
; (A)
  set port ha = 0 ; HTDレジスタの下位8ビットを選択
  wait 5ns ; delay
  set pin hwr = 0 ; ホスト・ライト・ストロブをアクティブにする

  input data ; データ・ファイルからデータ値を入力し "data" に代入
  set port hd = data & 0xFF ; HDポートから "data" の下位8ビットを入力する
  wait 100ns ; データを転送するまでウエイトする
  set pin hwr = 1 ; ホスト・ライト・ストロブをインアクティブにする
  wait 5ns ; delay
; (B)
; 上位8ビットについて上記(A) ~ (B)と同様の処理を繰り返す
  set port ha = 1 ; HTDレジスタの上位8ビットを選択
  wait 5ns ; delay
  set pin hwr = 0 ; ホスト・ライト・ストロブをアクティブにする

  set port hd= ( data >> 8 ) & 0xFF ; HDポートから "data" の上位8ビットを入力する
  wait 100ns ; データを転送するまでウエイトする
  set pin hwr = 1 ; ホスト・ライト・ストロブをインアクティブにする
  wait 5ns ; delay

  set pin hcs = 1 ; チップ・セレクトをインアクティブにして、入力を終了
  exit ( data == 0xFFFF ) ; "data" が0xFFFFなら、単語データ入力の終了
  wait 1ms ; 処理時間ウエイト
enddo ;
```

・終了

```
close input ; ファイルのクローズ
```

5.3.2 音声データの入力 (16 ビット・データ)

認識する音声データを，シリアル・インタフェースによって入力します。次に記述例を示します。

・準備

```
open input "wave.dat"           ; 入力ファイル指定 (符号付きデータ)
input format hex                 ; 入力ファイルのフォーマット指定 (16進数)
```

・入力処理 (16 ビット・データ)

```
do
  exit ( pin hre == 0 )          ; ホストI/Fから認識結果が出力されるまで繰り返す

  if pin sien1 == 0              ; シリアル入力イネーブルがインアクティブの場合
    set pin sck1 == 0            ;
    wait 4us                     ;
    set pin sck1 == 1            ; } 立ち上がりクロック (0 → 1) 入力
    wait 4us                     ;
    wait cond pin siak1 == 1     ; シリアル入力アクノリッジがアクティブになるまでウエイトする
    wait 5 ns                    ; delay
    set pin sien1 = 1            ; シリアル入力イネーブルをアクティブにする
  endif                          ;

  input port sil                  ; ファイルから値をシリアル入力する
  set pin sck1 == 0              ;
  wait 4us                       ;
  set pin sck1 == 1              ; } 立ち上がりクロック (0 → 1) 入力
  wait 4us                       ;
  wait cond pin siak1 == 0       ; シリアル入力アクノリッジがインアクティブになるまでウエイトする
  set pin sien1 = 0              ; シリアル入力イネーブルをインアクティブにする

  rept 15                         ; 15クロック・ウエイトする
    set pin sck1 == 0            ;
    wait 4us                     ;
    set pin sck1 == 1            ; } クロック入力
    wait 4us                     ;
  endrept                          ;

  if pin siak1 == 1              ; シリアル入力アクノリッジがアクティブなら，
    set pin sien1 = 1            ; シリアル入力イネーブルをアクティブにする
  endif                          ;

  set pin sck1 == 0              ;
  wait 4us                       ;
  set pin sck1 == 1              ; } 1 クロック入力
  wait 4us                       ;
enddo
```

・終了

```
close input                      ; ファイルのクローズ
```

5.3.3 認識結果の出力

認識した結果を、ホスト・インタフェースによって出力します。

・準備

```
open output "result.dat"      ; 出力ファイル指定
output format signed dec     ; 出力ファイルのフォーマット指定 (符号付き10進数)
```

・出力処理 (16 ビット・データ)

```
set pin hcs = 1              ; チップ・セレクトをインアクティブにする
set pin hrd = 1              ; ホスト・リード・ストロブもインアクティブにする

rept 14                      ; 認識結果として14ワードのデータをファイルに出力する
  wait cond pin hre == 0     ; ホストがHTD(out)に読み出し可能になるまでウェイトする
  wait cond pin hcs == 1     ; チップ・セレクトがインアクティブになるまでウェイトする

  set pin hcs = 0            ; チップ・セレクトをアクティブにして、出力を開始
; (A)
  set port ha = 0            ; HDTレジスタの下位8ビットを選択
  wait 5ns                  ; delay
  set pin hrd = 0            ; ホスト・リード・ストロブをアクティブにする
  wait 100ns                ; ウェイト

  set lowbyte = port hd & 0xff ; HDTから8ビットを読み出し "lowbyte" に代入
  set pin hrd = 1            ; ホスト・リード・ストロブをインアクティブにして、出力を終了する
  wait 5ns                  ; delay
; (B)
; 上位8ビットについて上記(A) ~ (B)と同様の処理を繰り返す
  set port ha = 1            ; HDTレジスタの上位8ビットを選択
  wait 5ns                  ; delay
  set pin hrd = 0            ; ホスト・リード・ストロブをアクティブにする
  wait 100ns                ; ウェイト

  output ((port hd&0xff)<<8) | lowbyte; HDTから8ビットを読み出し "lowbyte" とともにファイルへ出力

  set pin hrd = 1            ; ホスト・リード・ストロブをインアクティブにして、出力を終了する
  wait 5ns                  ; delay
  set pin hcs = 1            ; チップ・セレクトをインアクティブにして、出力を終了
endrept
```

・終了

```
close output                  ; ファイルのクローズ
```

付録 認識処理サンプル・ソース (sample.asm)

```

/*-----*/
/* File Information */
/*-----*/
/* Name      : sample.asm */
/* Type      : ASM Programming Language source code */
/* Version   : V1.00 */
/* Date      : 2001 March 16 */
/* CPU       : uPD77016 Family */
/* Compiler  : Atair uPD77016 Workbench */
/* About     : NEC uPD77016 Family Voice Recognition Middleware Library */
/*           : Sample Program */
/*-----*/
/* Copyright (C) NEC Corporation 2001 */
/* NEC CONFIDENTIAL AND PROPRIETARY */
/* All rights reserved by NEC Corporation. */
/* Use of copyright notice does not evidence publication */
/*-----*/

/*---- include file ----*/
#include "vrg_lib.h"

/*---- work area size (for 50 words) ----*/
#define STATICWORKSIZE 1552
#define DICSIZE        559

#define FRAME_LENGTH   128
#define RESULT_LENGTH  14

/*---- peripheral register ----*/
#define SDT1    0x3800    /* Serial data register 1 */
#define SST1    0x3801    /* Serial status register 1 */
#define HDT     0x3806    /* Host data register */
#define HST     0x3807    /* Host status register */

/*-----*/
/* Work Area / I/O Buffer */
/*-----*/
USER_XRAM   XRAMSEG
vin_stk:    ds 7          /* stack for serial interrupt */
voice_count: ds 1        /* sample count */
frame_count: ds 1        /* frame count */
libentry:   ds 3         /* temporary area for vrg_Init */

/* dictionary area */
dictionary: ds DICSIZE

/* voice buffer */
voice_buf0: ds FRAME_LENGTH /* 1st buffer */
voice_buf1: ds FRAME_LENGTH /* 2nd buffer */

/* voice buffer flag */
voice_buf_n: ds 1          /* 0:write to 1st buffer */
                          /* 1:write to 2nd buffer */

```

```

/* word buffer      */
adddic_temp:      ds 41          /* word buffer      */
adddic_temp_ct:  ds 1          /* character count */

/* static work area */
/* Example in the case of not sharing static work area with __VRG_YRAM5 */
/*
USER_YRAM   YRAMSEG
staticwork: ds STATICWORKSIZE
*/

/*****/
/* Vector Table */
/*****/
Vct_Ix  imseg at 0x0200
    nop                ; reset
    jmp main           ;
    nop                ;
    nop                ;
    nop                ; not used
    reti              ;
    nop                ;
    nop                ;
    nop                ; not used
    reti              ;
    nop                ;
    nop                ;
    nop                ; not used
    reti              ;
    nop                ;
    nop                ;
    nop                ; INT0
    reti              ;
    nop                ;
    nop                ;
    nop                ; INT1
    reti              ;
    nop                ;
    nop                ;
    nop                ; INT2
    reti              ;
    nop                ;
    nop                ;
    nop                ; INT3
    reti              ;
    nop                ;
    nop                ;
    nop                ; SI#1
    call voice_intrpt ;
    reti              ;
    nop                ;
    nop                ; SO#1
    reti              ;
    nop                ;
    nop                ;
    nop                ; SI#2
    reti              ;
    nop                ;
    nop                ;
    nop                ; SO#2
    reti              ;

```

```

nop                ;
nop                ;
nop                ; HOST IN
reti               ;
nop                ;
nop                ;
nop                ; HOST OUT
reti               ;
nop                ;
nop                ;
nop                ; not used
reti               ;
nop                ;
nop                ;
nop                ; not used
reti               ;
nop                ;
nop                ;
nop                ;

/*****/
/* Sample Source Code */
/*****/
Smp_Ix   imseg
main:
    R0L=0          ;
    *HST:X = R0L  ;
    R0L=0x200     ;
    *SST1:X = R0L ;

    DP0 = 0       ;
    DP1 = 0       ;
    DP2 = 0       ;
    DP3 = 0       ;
    DP4 = 0       ;
    DP5 = 0       ;
    DP6 = 0       ;
    DP7 = 0       ;
    DN0 = 0       ;
    DN1 = 0       ;
    DN2 = 0       ;
    DN3 = 0       ;
    DN4 = 0       ;
    DN5 = 0       ;
    DN6 = 0       ;
    DN7 = 0       ;
    DMX = 1       ;
    DMY = 1       ;
    CLR ( R0 )    ;
    CLR ( R1 )    ;
    CLR ( R2 )    ;
    CLR ( R3 )    ;
    CLR ( R4 )    ;
    CLR ( R5 )    ;
    CLR ( R6 )    ;
    CLR ( R7 )    ;

/* clear memory */
    R0L = 0       ;
    *adddic_temp_ct:X = R0L ;

/* disable interrupt */

```

```

    R0L=EIR                ;
    R0=R0|0x8000          ;
    EIR=R0L               ; disable interrupt
    NOP                   ;
    NOP                   ;

/* vrg_InitDic */
    DP0 = dictionary      ;
    R0L = DICSIZE         ;
    CALL vrg_InitDic     ; CALL vrg_InitDic

/* input word */
    CLR ( R0 )           ;
poll:
    R0L = *HST:X         ;
    R0 = R0 & 1         ;
    IF ( R0 != 0 ) JMP poll ; host I/F status check

    R0L = *HDT:X         ; read data
    R1 = R0 - 0xFFFF    ;
    IF ( R1 == 0 ) JMP words_end ;
    CLR ( R1 )          ;
    CLR ( R2 )          ;
    R1L = adddic_temp   ;
    R2L = *adddic_temp_ct:X ;
    R1 = R1 + R2        ;
    DP0 = R1L           ;
    NOP                 ;
    *DP0 = R0L          ;
    R2 = R2 + 1         ;
    *adddic_temp_ct:X = R2L ;
    IF ( R0 == 0 ) JMP input_word ;
    JMP poll            ;

input_word:
    DP0 = dictionary    ;
    DP1 = adddic_temp   ;
    CALL vrg_AddDic     ; CALL vrg_AddDic
    IF ( R0 < 0 ) JMP error ;
    CLR ( R0 )          ;
    *adddic_temp_ct:X = R0L ;
    JMP poll            ;

words_end:
/* enable interrupt */
    R0L = SR            ;
    R0 = R0 & 0xFFEF   ;
    SR = R0L           ; enable int SI
    NOP                ;
    NOP                ;

start_recog:
    DP0 = libentry     ;
    R0L = dictionary   ;
    *DP0++ = R0L       ;
    R0L = VRG_YRAM5    ;
/* Example in the case of not sharing static work area with __VRG_YRAM5 */
/* R0L = staticwork ; */
    *DP0++ = R0L       ;
    R0L = STATICWORKSIZE ;
    *DP0++ = R0L       ;

```

```

DP0 = libentry                ;
CALL vrg_Init                 ; CALL vrg_Init
IF ( R0 != 0 ) JMP error      ;

/* clear memory */
ROL = 0                        ;
*voice_count:X = ROL          ;
*frame_count:X = ROL          ;
*voice_buf_n:X = ROL          ;

/* enable interrupt */
FINT                           ; clear interrupt
ROL = EIR                       ;
R0 = R0 & 0x7FFF                ;
EIR = ROL                       ; enable interrupt
NOP                             ;
NOP                             ;

host_poll:
ROL = *HST:X                    ;
R0 = R0 & 2                      ;
IF ( R0 != 0 ) JMP host_poll    ; host I/F status check

ROL = 0                          ; send zero
*HDT:X = ROL                     ; "ready for recognition"

IDLE_LOOP:
NOP                             ; HALT or NOP

check_flag:
CLR ( R0 )                       ;
ROL = *frame_count:X            ; check frame count
IF ( R0 > 0 ) JMP run_recog     ;
JMP IDLE_LOOP                   ;

run_recog:
R0 = R0 - 1                      ; decrement frame_count
*frame_count:X = ROL            ;

/* set voice buffer address */
ROL = voice_buf0                 ;
CLR ( R1 )                       ;
R1L = *voice_buf_n:X            ;
IF ( R1 != 0 ) JMP call_recog   ;
ROL = voice_buf1                 ;

call_recog:
DP0 = ROL                         ;
CALL vrg_Recog                   ; CALL vrg_Recog
IF ( R0 == 0 ) CALL recog_fin    ;
IF ( R0 < 0 ) JMP error          ;
JMP IDLE_LOOP                   ;

recog_fin:
/* disable interrupt */
ROL=EIR                           ;
R0=R0|0x8000                       ;
EIR=R0L                             ; disable interrupt
NOP                                 ;
NOP                                 ;

```

```

/* send recognition result */
    DP0 = vrg_result                ;
    CLR ( R0 )                      ;
    LOOP RESULT_LENGTH {           ;
ho_wait:
    R0L = *HST:X                    ;
    R0 = R0 & 2                    ;
    IF ( R0 != 0 ) JMP ho_wait      ; host I/F status check

    R0L = *DP0++                    ;
    *HDT:X = R0L                    ; send data
    NOP                             ;
    }                               ;
    JMP start_recog                 ;

error:
    NOP                             ; CHECK ERROR CODE (R0 REGISTER)
    JMP error                       ;

/* int SI handler */
voice_intrpt:
    *vin_stk+0:X = R0L              ;
    *vin_stk+1:X = R0H              ;
    *vin_stk+2:X = R0E              ;
    *vin_stk+3:X = R1L              ;
    *vin_stk+4:X = R1H              ;
    *vin_stk+5:X = R1E              ;
    R0L = DP0                       ;
    *vin_stk+6:X = R0L              ;

/* set voice buffer address */
    CLR ( R0 )                      ;
    CLR ( R1 )                      ;
    R0L = voice_buf0                ;
    R1L = *voice_buf_n:X            ;
    IF ( R1 == 0 ) JMP store_voice  ;
    R0L = voice_buf1                ;

store_voice:
    R1L = *voice_count:X            ;
    R0 = R0 + R1                    ;
    DP0 = R0L                       ;
    R0L = *SDT1:X                   ; read data
    *DP0 = R0L                      ;
    R1 = R1 + 1                     ;
    R0 = R1 - FRAME_LENGTH          ;
    IF ( R0 < 0 ) JMP voice_int_end ;

/* increment frame_count */
    CLR ( R0 )                      ;
    R0L = *frame_count:X            ;
    R0 = R0 + 1                     ;
    *frame_count:X = R0L            ;

/* switch buffer */
    CLR ( R1 )                      ;
    R1L = *voice_buf_n:X            ;
    R1 = R1 + 1                     ;
    R1 = R1 & 1                      ;

```

```
*voice_buf_n:X = R1L ;
CLR ( R1 ) ;

voice_int_end:
*voice_count:X = R1L ;
R0L = *vin_stk+6:X ;
DPO = R0L ;
R1E = *vin_stk+5:X ;
R1H = *vin_stk+4:X ;
R1L = *vin_stk+3:X ;
R0E = *vin_stk+2:X ;
R0H = *vin_stk+1:X ;
R0L = *vin_stk+0:X ;
RET ;

END
```

[メ モ]

[メ 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 デバイス ホームページ】

NECエレクトロニクスデバイ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/>

アンケート記入のお願い

お手数ですが、このドキュメントに対するご意見をお寄せください。今後のドキュメント作成の参考にさせていただきます。

[ドキュメント名] μ SAP77016-B10 ユーザーズ・マニュアル (U15518JJ1V0UM00 (第1版))

[お名前など](さしつかえのない範囲で)

- 御社名(学校名, その他) ()
- ご住所 ()
- お電話番号 ()
- お仕事の内容 ()
- お名前 ()

1. ご評価(各欄に をご記入ください)

項目	大変良い	良い	普通	悪い	大変悪い
全体の構成					
説明内容					
用語解説					
調べやすさ					
デザイン, 字の大きさなど					
その他()					
()					

2. わかりやすい所(第 章, 第 章, 第 章, 第 章, その他)
理由 []

3. わかりにくい所(第 章, 第 章, 第 章, 第 章, その他)
理由 []

4. ご意見, ご要望

5. このドキュメントをお届けしたのは
NEC 販売員, 特約店販売員, その他 ()

ご協力ありがとうございました。
下記あてに FAX で送信いただくか, 最寄りの販売員にコピーをお渡ししてください。