

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。

アプリケーション・ノート

78K0R/Kx3

サンプル・プログラム (I²Sバス・インタフェース)

オーディオCODECとのI²Sバス・インタフェース編

この資料は、サンプル・プログラムの動作概要や使用方法、および接続されたオーディオCODECを使用するためのI²Sバス・インタフェースの設定方法や活用方法について説明したものです。このサンプル・プログラムはI²Sバス・インタフェースにて、オーディオCODECとオーディオ・データの送受信をし、マイク入力した音声の録音と再生を行うものです。また、オーディオCODEC内のレジスタ設定にI²Cバス・インタフェース、およびオーディオ・データのEEPROMへのリード/ライトにはSPIインタフェースを使用します。

対象デバイス

78K0R/KE3マイクロコントローラ
 78K0R/KF3マイクロコントローラ
 78K0R/KG3マイクロコントローラ
 78K0R/KH3マイクロコントローラ
 78K0R/KJ3マイクロコントローラ

目次

第1章 概要 ...	3
第2章 回路図 ...	6
2.1 回路図 ...	6
2.2 周辺ハードウェア ...	7
第3章 ソフトウェアについて ...	8
3.1 ファイル構成 ...	8
3.2 使用する内蔵周辺機能 ...	9
3.3 使用する周辺の初期設定と動作概要 ...	11
3.4 フロー・チャート ...	13
3.5 I ² Sバス・インタフェースのフォーマット ...	21
第4章 設定方法について ...	23
4.1 I ² Sバス・インタフェースを使用するための設定 ...	24
4.2 変数と定数の定義 ...	41
4.3 使用する周辺の初期設定 ...	45
4.4 メイン処理 ...	59
4.5 INTTM00割り込み処理 ...	65
4.6 INTCSI00割り込み処理 ...	67
4.7 INTCSI10割り込み処理 ...	72
4.8 INTDMA0割り込み処理 ...	77
4.9 INTDMA1割り込み処理 ...	80
4.10 I ² Cバス・インタフェース・ライト処理 ...	82
第5章 関連資料 ...	85
付録A プログラム・リスト ...	86
付録B 改版履歴 ...	138

資料番号 U19514JJ1V0AN00 (第1版)

発行年月 November 2008 NS

- 本資料に記載されている内容は2008年11月現在のものです、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いません。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、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う。
- (2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

M8E0710J

第1章 概要

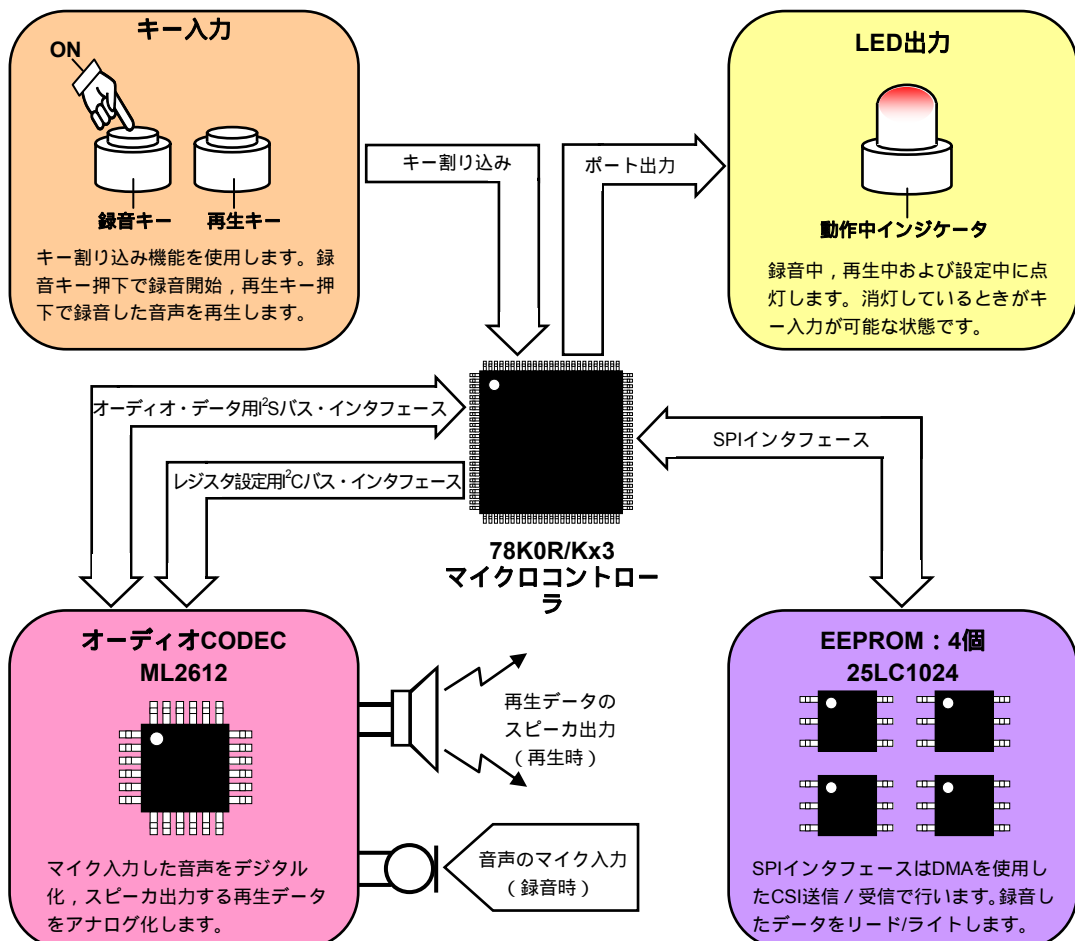
このサンプル・プログラムは、I²Sバス・インタフェースを応用し、オーディオCODECとのオーディオ・データの送受信を行うものです。I²Sバス・インタフェースのLチャンネルかRチャンネルを選択するためのLRクロック (LRCLK) をタイマ・アレイ・ユニット (TAU) のインターバル・タイマ・モードを使用し、TO00より出力します。データを送受信するためのクロック (BCLK) の出力、データの受信 (SDOUT) およびデータの送信 (SDIN) をシリアル・アレイ・ユニット0 (SAU0) のチャンネル0のCSI00を使用します。

OKIセミコンダクタ製のオーディオCODEC ML2612を使用し、オーディオCODEC内のレジスタ設定にI²Cバス・インタフェースを使用し、オーディオ・データにはI²Sバス・インタフェースを使用します。録音キー押下時はオーディオCODECによるマイク入力した音声データを受信し、EEPROMへデータを保存します。また、再生キー押下時は録音したデータをオーディオCODECへ送信し、スピーカ出力を行います。EEPROMとのSPIインタフェースはシリアル・アレイ・ユニット0 (SAU0) のチャンネル2のCSI10を使用し、DMAコントローラによりCSI送信 / 受信を続けて行います。録音中、再生中の状態は、動作中インジケータ用LEDをP72で出力し、表示させます。

また、このサンプル・プログラムでは、I²Sバス・インタフェースの応用のためオーディオCODEC用のICを使用し、音声の録音、再生をしています。本マイコン単体においてもADPCM-SPライブラリを使用して実現することができます。

【動作概要】

サンプル・プログラムの動作概要を示します。



このサンプル・プログラムの処理の概要を以下に示します。

(1) 使用する周辺の初期設定の主な内容

使用する周辺の初期設定の主な内容は、次のとおりです。

割り込みの禁止

CPU / 周辺ハードウェア・クロック周波数をX1発振クロック (20 MHz使用) に設定

ポートの設定

オーディオCODECへの設定

- ・オーディオCODECへのクロック供給
- ・プログラム内ウエイト用のタイマ設定
- ・オーディオCODECへのリセット信号出力
- ・レジスタ設定用のI²Cインタフェース設定
- ・オーディオCODEC内のレジスタ設定にてシステムON

オーディオCODECのオーディオ・データ用I²Sバス・インタフェース設定

- ・TO00出力 (16 kHz) を使用し、LRCLKの出力設定
- ・CSI00の設定

EEPROMのSPIインタフェース設定

- ・データ送信 / 受信にCSI10の設定
- ・受信にDMA0, 送信にDMA1を設定
- ・EEPROMの全消去 (オールOFFH)

キーの取り込み開始

割り込みの許可

(2) メイン処理の主な内容

メイン処理の主な内容は、次のとおりです。

- ・キー処理
- ・録音開始設定
- ・再生開始設定
- ・録音終了設定
- ・再生終了設定

(3) INTTM00割り込み処理 (INTTM00使用, LRCLKとBCLK, SDOUTおよびSDIN同期用) の主な内容

INTTM00割り込み処理では、LRCLKとBCLK, SDOUTおよびSDINを同期させるためにI²Sバス・インタフェース動作を開始します。

(4) INTCSI00割り込み処理 (INTCSI00使用, CSI送信/受信用) の主な内容

INTCSI00割り込み処理の主な内容は、次のとおりです。

- I²Sバス・インタフェースでのデータ送信/受信
- 受信データの保存
- EEPROMへのライト開始
- EEPROMへのリード開始

(5) INTCSI10割り込み処理 (INTCSI10使用, CSI送信/受信用) の主な内容

INTCSI10割り込み処理の主な内容は、次のとおりです。

- EEPROMへの命令バイト送信
- EEPROMへの24ビット・アドレス送信
- DMA0, DMA1を使用したEEPROMからのデータ受信開始
- DMA1を使用したEEPROMへのデータ送信開始

(6) INTDMA0割り込み処理 (INTDMA0使用, CSI受信用) の主な内容

INTDMA0割り込み処理の主な内容は、次のとおりです。

- EEPROMからのリード開始
- I²Sバス・インタフェース動作開始
- EEPROMの選択
- I²Sバス・インタフェース動作停止

(7) INTDMA1割り込み処理 (INTDMA1使用, CSI送信用) の主な内容

INTDMA1割り込み処理の主な内容は、次のとおりです。

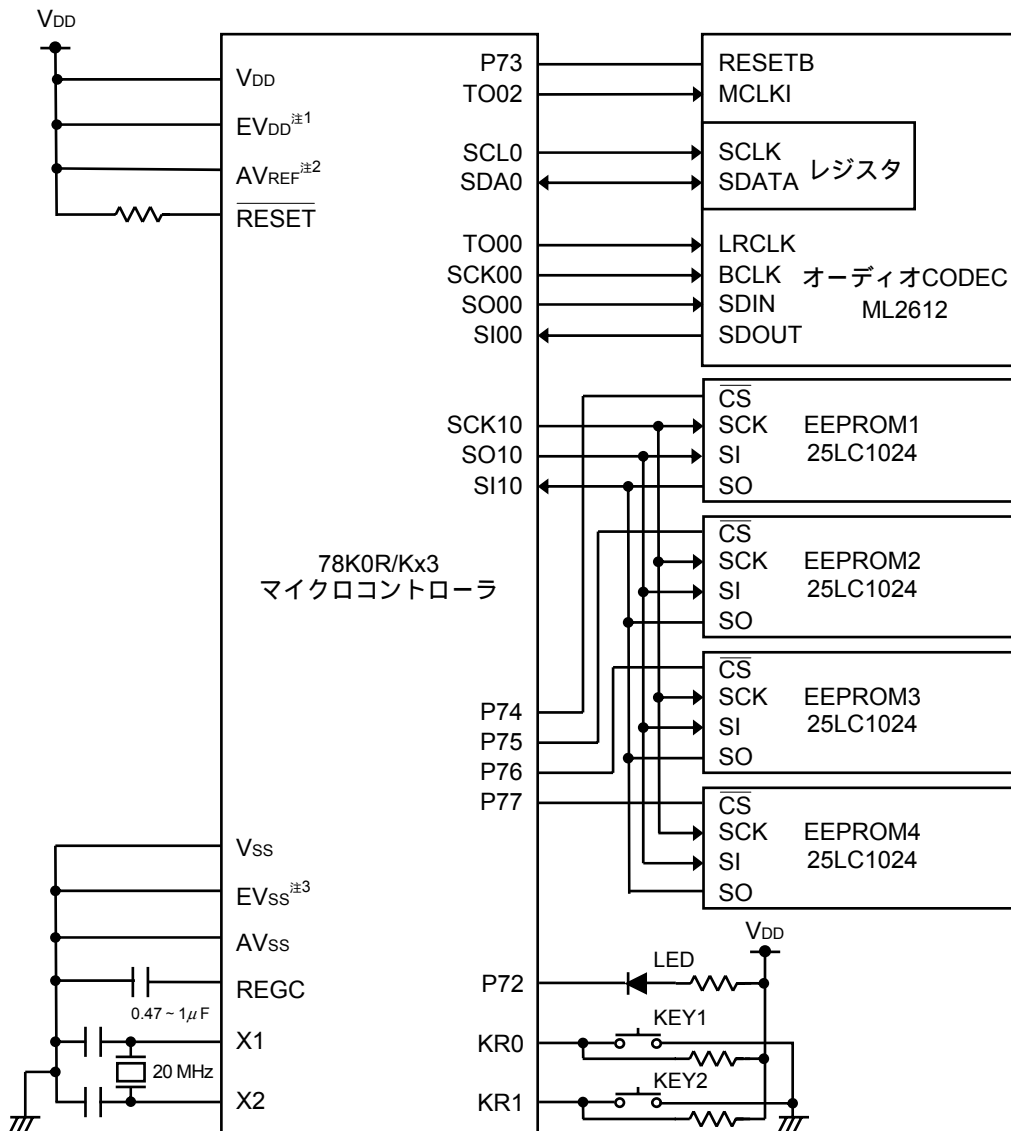
- EEPROMの選択
- I²Sバス・インタフェース動作停止

第2章 回路図

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図および周辺ハードウェアを説明します。

2.1 回路図

回路図を次に示します。



注1. 78K0R/KG3, 78K0R/KH3, 78K0R/KJ3マイクロコントローラはEV_{DD0}とEV_{DD1}になります。

2. 78K0R/KF3, 78K0R/KG3, 78K0R/KH3, 78K0R/KJ3マイクロコントローラはAV_{REF0}とAV_{REF1}になります。

3. 78K0R/KG3, 78K0R/KH3, 78K0R/KJ3マイクロコントローラはEV_{SS0}とEV_{SS1}になります。

注意1. 2.7 V V_{DD} 3.6 Vの電圧範囲で使用してください。

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

3. EV_{DD}は、V_{DD}と同電位にしてください。

4. REGCはコンデンサ (0.47 ~ 1μF) を介し、V_{SS}に接続してください。

5. 回路図中の端子以外の未使用のポート機能端子はすべて出力ポートのため、オープンにしてください。

2.2 周辺ハードウェア

使用する周辺ハードウェアを次に示します。

(1) OKIセミコンダクタ製オーディオCODEC ML2612

音声のマイク入力および音声データのスピーカ出力用として使用します。

(2) EEPROM 25LC1024 (EEPROM1, EEPROM2, EEPROM3, EEPROM4)

音声データの保存用に使用します。

(3) キー (KEY1, KEY2)

録音開始 / 再生開始制御の入力として、キーを使用します。

(4) LED

録音中、再生中などキー入力無効時の状態表示の出力として、LEDを使用します。

使用する端子機能を次に示します。



外部デバイス接続時の端子機能		兼用端子
名称	機能	
P73	オーディオCODEC ML2612へのリセット	KR3
TO02	オーディオCODEC ML2612へのマスタクロック供給	P17/TI02
SCL0	オーディオCODEC ML2612のレジスタへのI ² Cバス・インタフェース用クロック出力	P60
SDA0	オーディオCODEC ML2612のレジスタへのI ² Cバス・インタフェース用データ入出力	P61
TO00	オーディオCODEC ML2612のI ² Sバス・インタフェース用LRクロック出力	P01
SCK00	オーディオCODEC ML2612のI ² Sバス・インタフェース用ビット・クロック出力	P10/SCL10
SI00	オーディオCODEC ML2612のI ² Sバス・インタフェース用データ入力	P11/RxD0
SO00	オーディオCODEC ML2612のI ² Sバス・インタフェース用データ出力	P12/TxD0
P74-P77	EEPROM 25LC1024へのチップ・セレクト出力 (EEPROMを4個使用します)	P74-P77
SCK10	EEPROM 25LC1024へのSPIインタフェース用クロック出力	P04/SCL10
SI10	EEPROM 25LC1024へのSPIインタフェース用データ入力	P03/RxD1/SDA10
SO10	EEPROM 25LC1024へのSPIインタフェース用データ出力	P02/TxD1
P72	動作状態表示用LED出力	KR2
KR0	再生キー入力	P70
KR1	録音キー入力	P71

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)	マイコンのハードウェア初期化処理、メイン処理と 割り込み処理のソース・ファイル	注	注
main.c (C言語版)			
I2sApplication.prw	統合開発環境 PM+用ワーク・スペース・ファイル		
I2sApplication.prj	統合開発環境 PM+用プロジェクト・ファイル		

注 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。

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のTO00出力：
オーディオCODEC ML2612へのI²Sバス・インタフェース用のLRクロック出力として16 kHz出力で使用します。
- ・ タイマ・アレイ・ユニット0 (TAU0) のチャンネル1のインターバル・タイマ：
プログラム内のウエイト時間の作成用として5 msのインターバル・タイマ・モードで使用します。
- ・ タイマ・アレイ・ユニット0 (TAU0) のチャンネル2のTO02出力：
オーディオCODEC ML2612へのマスタクロック供給用として10 MHz出力で使用します。
- ・ シリアル・アレイ・ユニット0 (SAU0) のチャンネル0のCSI00：
オーディオCODEC ML2612へのI²Sバス・インタフェース用のビット・データの入出力として転送速度833 kHzで使用します。
- ・ シリアル・アレイ・ユニット0 (SAU0) のチャンネル2のCSI10：
EEPROM 25LC1024とのSPIインタフェース用のデータ入出力として転送速度2.5 MHzで使用します。
- ・ DMAコントローラのチャンネル0 (DMA0) とチャンネル1 (DMA1)：
EEPROM 25LC1024とのSPIインタフェース用として使用するCSI10の送信 / 受信に使用します。
- ・ シリアル・インタフェースIIC0：
オーディオCODEC ML2612のレジスタ設定に使用します。I²Cバス・インタフェースでのデータ入出力として転送速度208 kHzで使用します。
- ・ キー割り込み：
録音開始キーと再生開始キーの入力用として使用

・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	
P73	オーディオCODEC ML2612へのリセット	KR3
TO02	オーディオCODEC ML2612へのマスタクロック供給	P17/TI02
SCL0	オーディオCODEC ML2612のレジスタへのI ² Cバス・インタフェース用クロック出力	P60
SDA0	オーディオCODEC ML2612のレジスタへのI ² Cバス・インタフェース用データ入出力	P61
TO00	オーディオCODEC ML2612のI ² Sバス・インタフェース用LRクロック出力	P01
SCK00	オーディオCODEC ML2612のI ² Sバス・インタフェース用ビット・クロック出力	P10/SCL10
SI00	オーディオCODEC ML2612のI ² Sバス・インタフェース用データ入力	P11/RxD0
SO00	オーディオCODEC ML2612のI ² Sバス・インタフェース用データ出力	P12/TxD0
P74-P77	EEPROM 25LC1024へのチップ・セレクト出力（EEPROMを4個使用します）	P74-P77
SCK10	EEPROM 25LC1024へのSPIインタフェース用クロック出力	P04/SCL10
SI10	EEPROM 25LC1024へのSPIインタフェース用データ入力	P03/RxD1/SDA10
SO10	EEPROM 25LC1024へのSPIインタフェース用データ出力	P02/TxD1
P72	動作状態表示用LED出力	KR2
KR0	再生キー入力	P70
KR1	録音キー入力	P71

3.3 使用する周辺の初期設定と動作概要

このサンプル・プログラムでは、使用する周辺の初期設定にて、クロック周波数の選択や、I²Sバス・インタフェースで使用するタイマ・アレイ・ユニットとシリアル・アレイ・ユニットの設定、I²Cバス・インタフェースで使用するシリアル・インタフェースIIC0の設定、SPIインタフェースで使用するシリアル・アレイ・ユニットの設定、プログラム内のウエイト用として使用するタイマ・アレイ・ユニットの設定、録音/再生キーの入力用としてキー割り込みの設定などを行います。

使用する周辺の初期設定完了後、メイン処理へ処理が移行します。

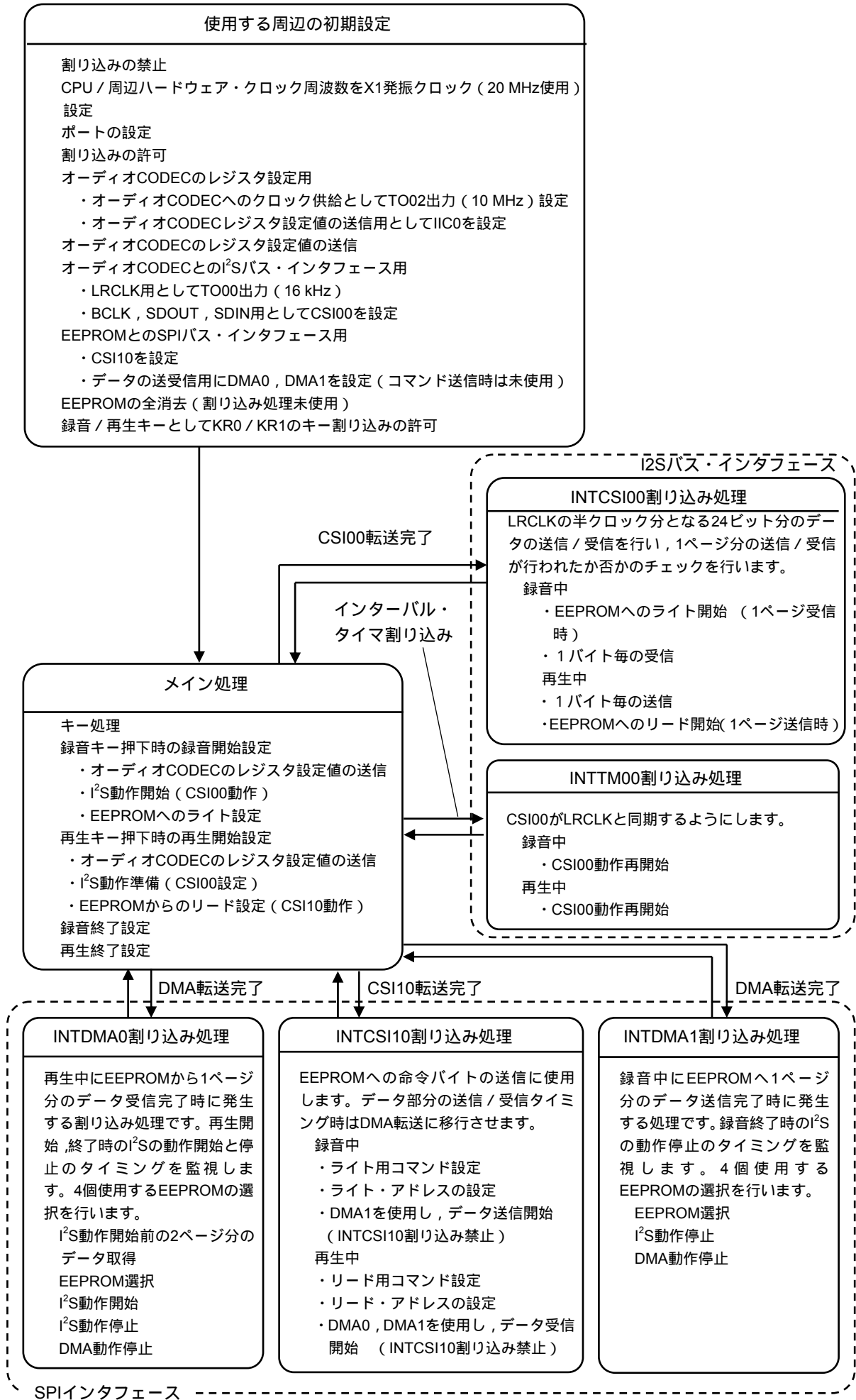
録音キーの入力があれば、録音を開始します。録音開始時は、まず録音可能となるようにI²CインタフェースでオーディオCODECのレジスタの設定を行います。その後、マイク入力された音声は、オーディオCODECからI²Sバス・インタフェースでオーディオ・データとして受信できます。受信したデータは、1ページ^{*}分のデータを受信するたびにSPIインタフェースでEEPROMへ送信します。EEPROMへ送信したデータがEEPROMの4個分となったとき送信を終了し、オーディオCODECへ録音停止となるように、I²CインタフェースでオーディオCODECのレジスタを設定します。

再生キーの入力があれば、録音したデータの再生を開始します。再生開始時は、まず再生可能となるようにI²CインタフェースでオーディオCODECのレジスタの設定を行います。その後、SPIインタフェースでEEPROMから2ページ分の再生データを受信し、オーディオCODECへI²Sバス・インタフェースでオーディオ・データとして送信することで、再生データをスピーカ出力します。再生データは、1ページ^{*}分のデータを送信するたびにSPIインタフェースでEEPROMから受信します。EEPROMから受信したデータがEEPROMの4個分となったとき受信を終了し、オーディオCODECへ再生停止となるように、I²CインタフェースでオーディオCODECのレジスタを設定します。

録音中、再生中は動作中を示す表示として、P72に接続された動作中インジケータ（LED）を点灯させます。

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。

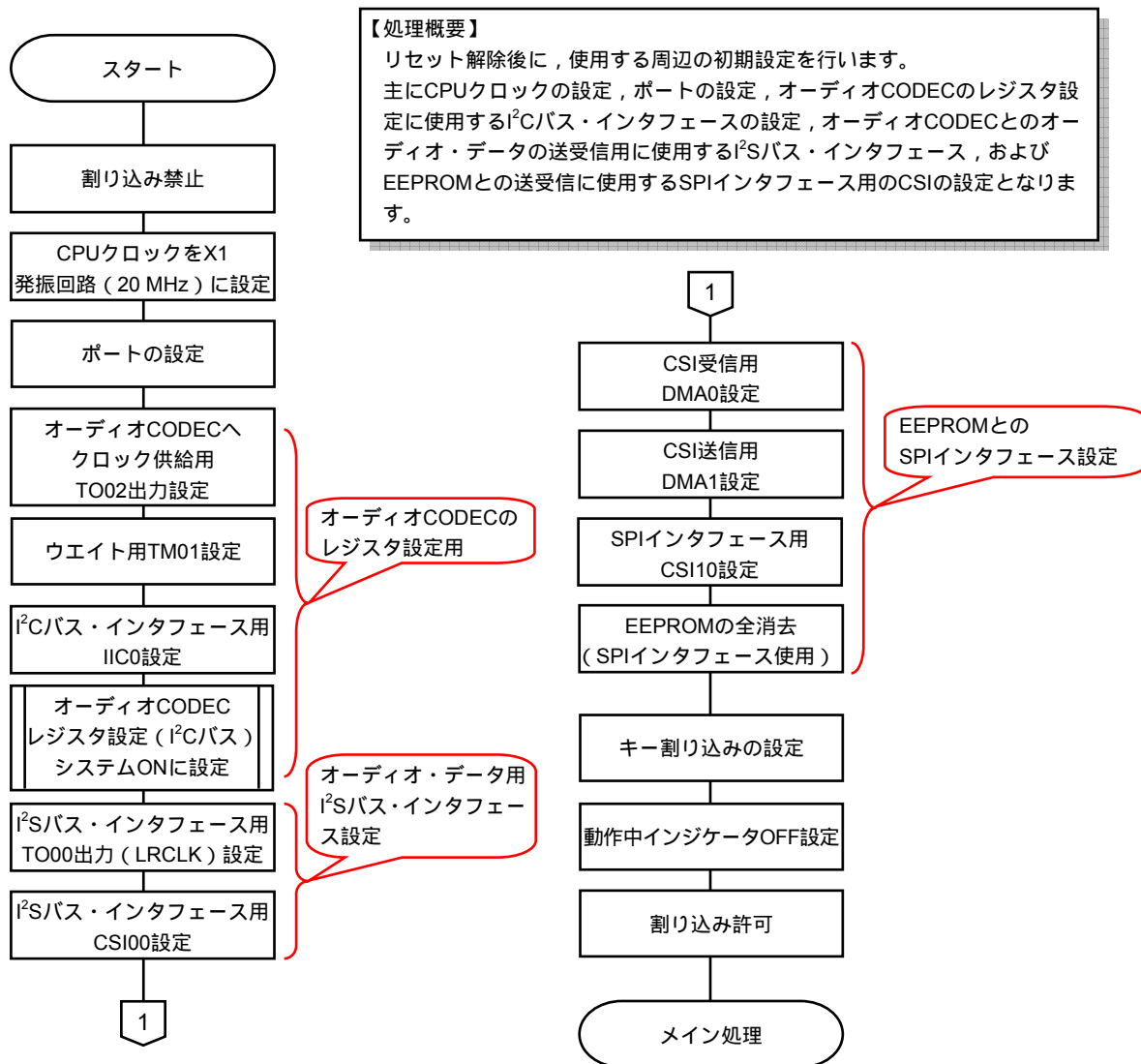
注 LRクロックの半クロックの範囲内に24ビットのオーディオ・データが含まれますので、LRクロックの1クロックで計6バイトのオーディオ・データが送受信されます。EEPROMの1ページのサイズは、256バイトですが、このサンプル・プログラムでは、LRクロック42クロック分のオーディオ・データとなる252バイトをプログラム内で使用する1ページとしています。



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。

<リセット解除後の使用する周辺の初期設定>

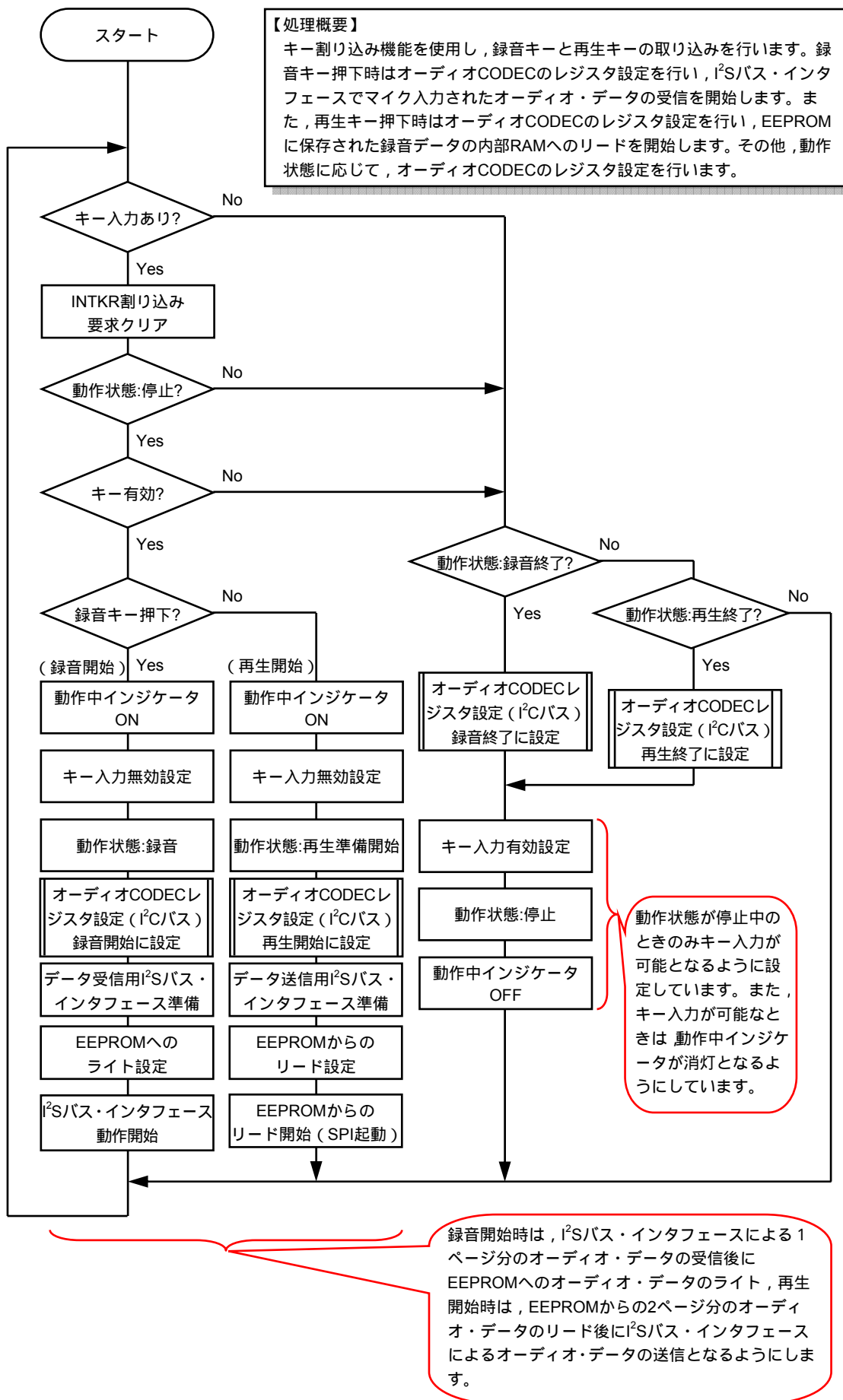


注意 オプション・バイトは、RA78K0Rのリンク・オプションにて設定してください。設定の仕方については、RA78K0R アセンブラ・パッケージ ユーザーズ・マニュアルを参照してください。

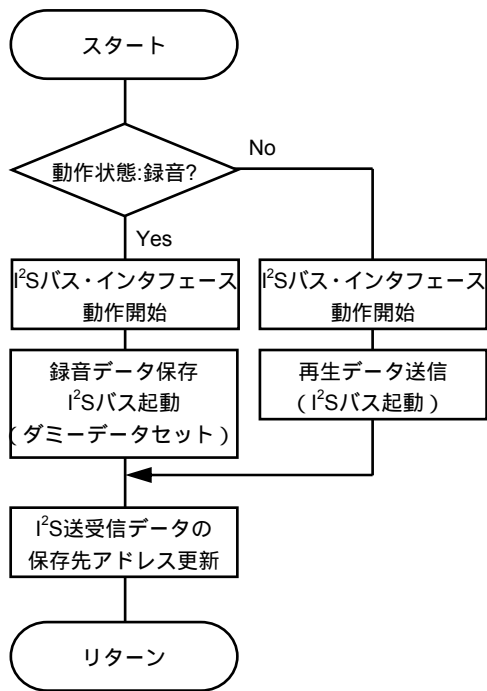
オプション・バイトは、次の内容が設定されます。

- ・ウォッチドッグ・タイマの動作
- ・リセット解除時（電源立ち上げ時）のLVIの設定
- ・オンチップ・デバッグの動作制御

<メイン処理>

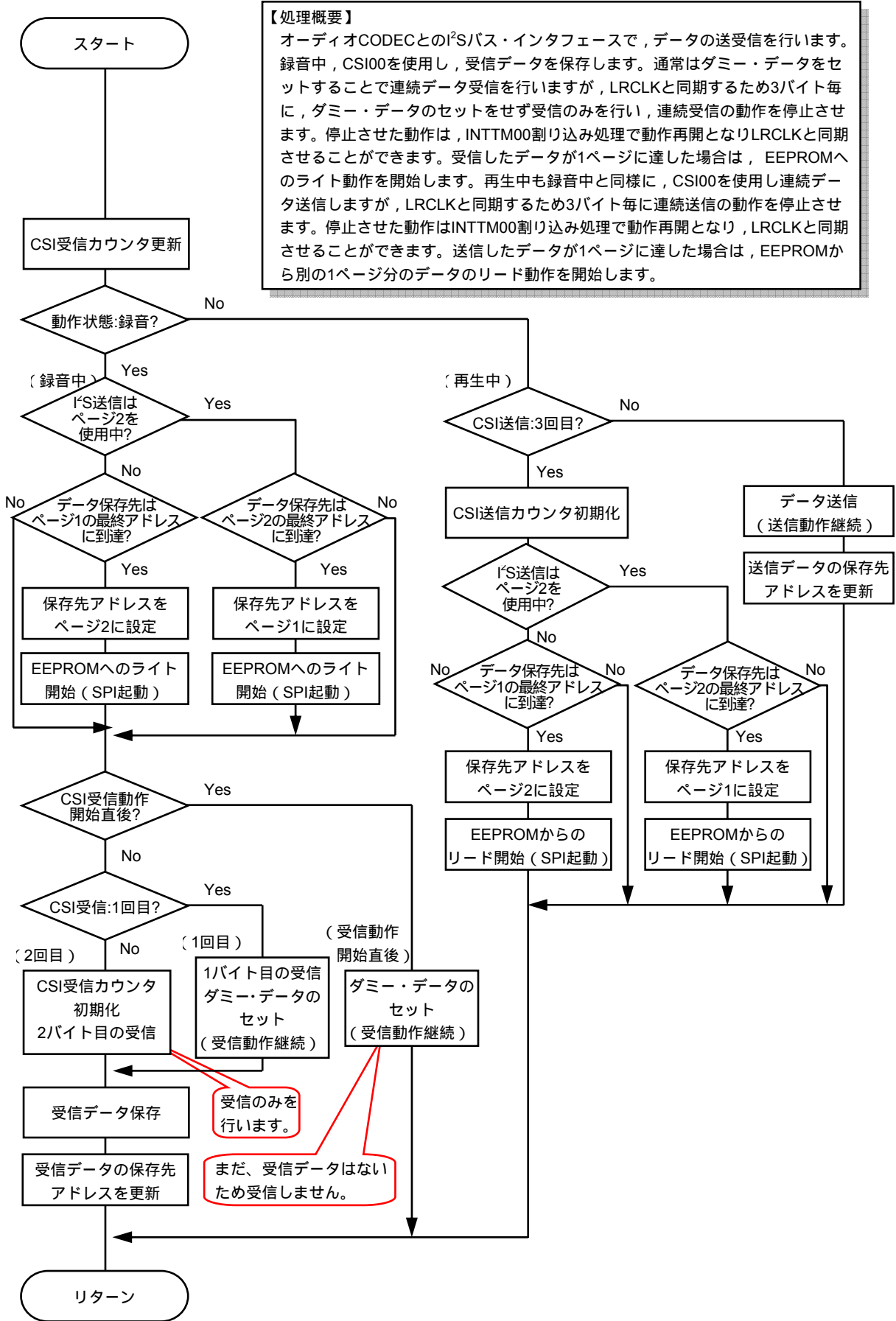


< LRCLK同期用INTTM00割り込み処理 >

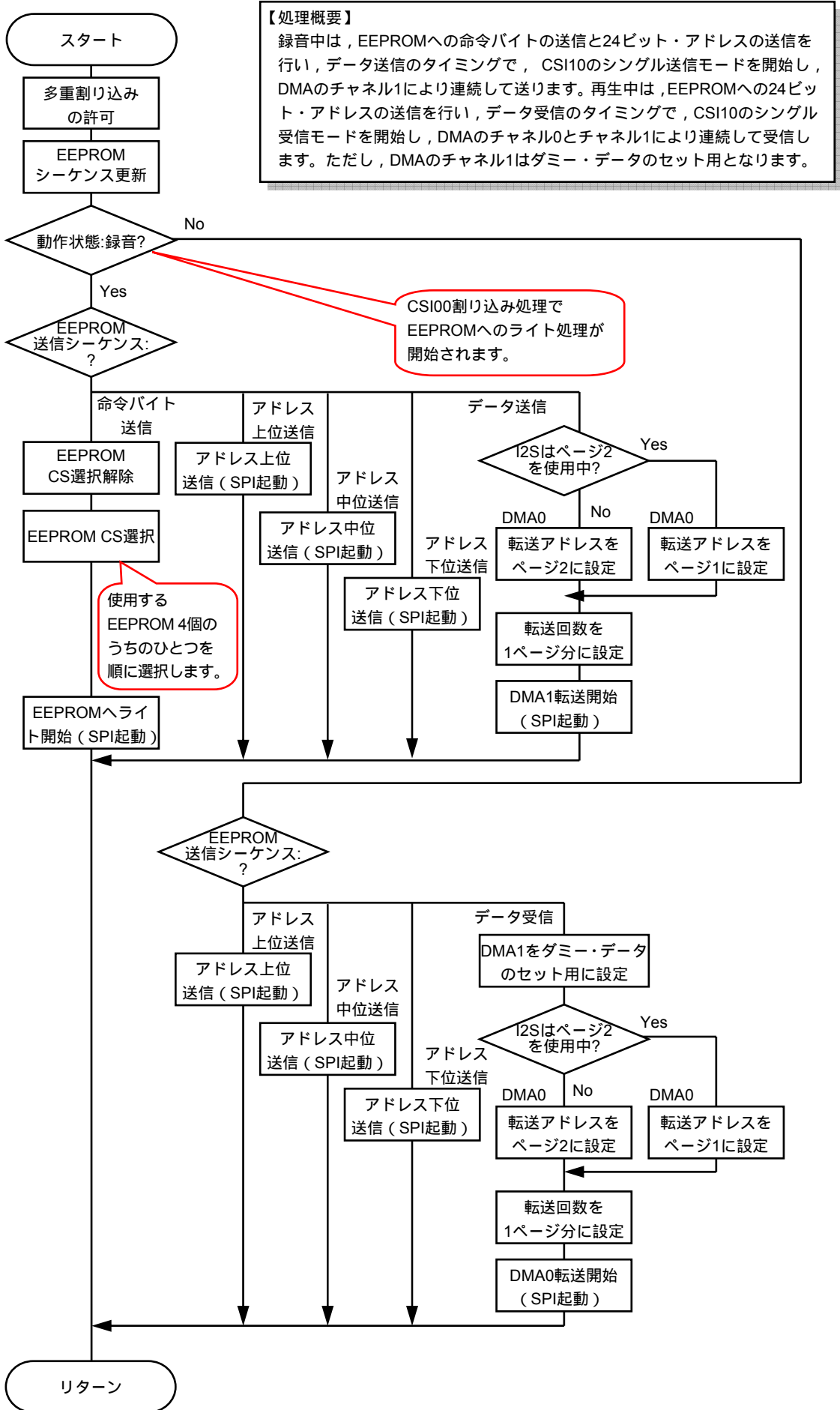


【処理概要】
 I²Sバス・インタフェースのTO00出力によるLRCLKとデータ送信 / 受信で使用するCSI00によるBCLK, SDOUTおよびSDINの同期を行います。INTCSI00割り込み処理で一度、動作を停止させたCSI00の動作をLRCLKのハイ / ロウ変化のタイミングで再開させ同期させます。

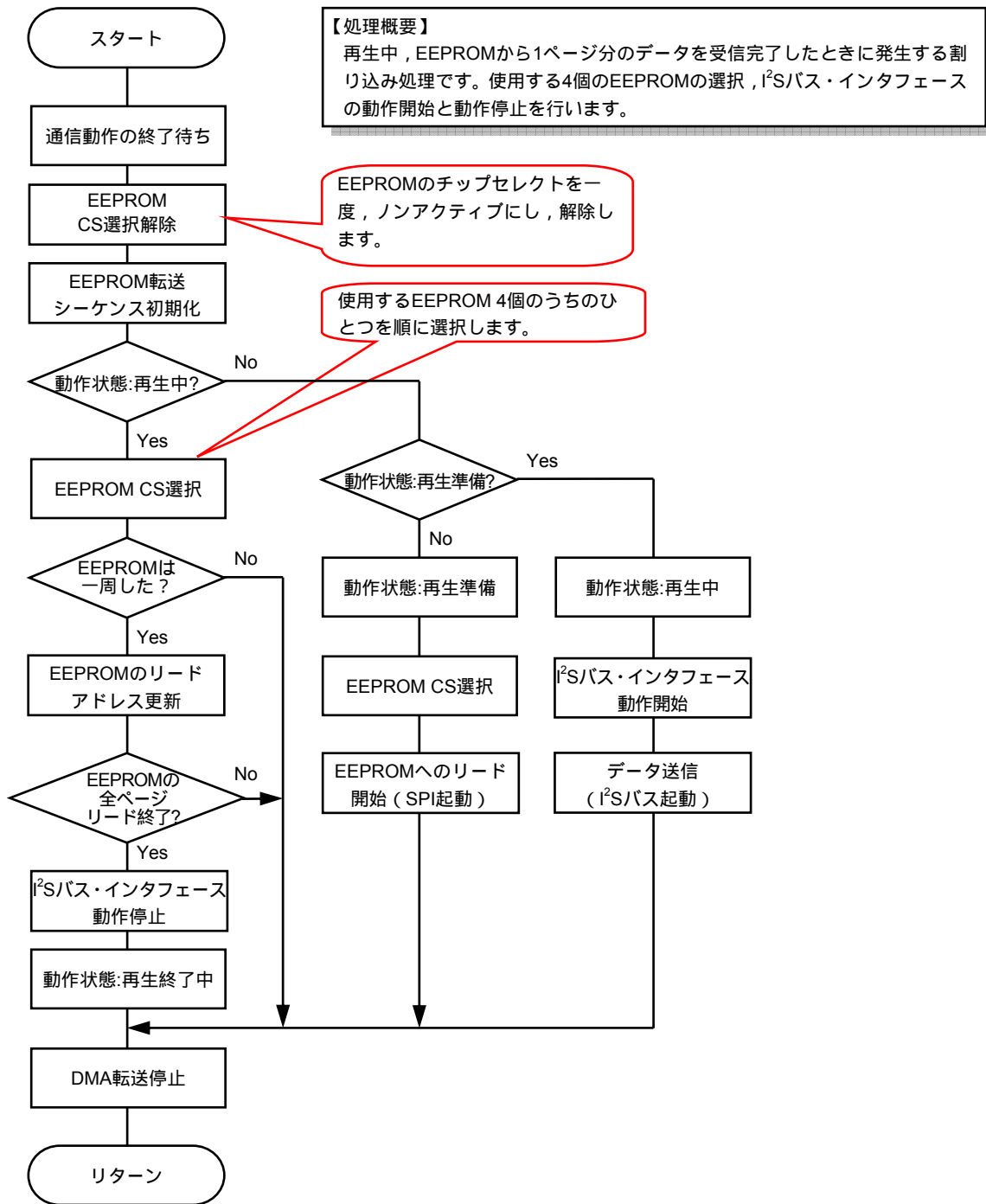
< I²Sバス・インタフェース用INTCSI00割り込み処理 >



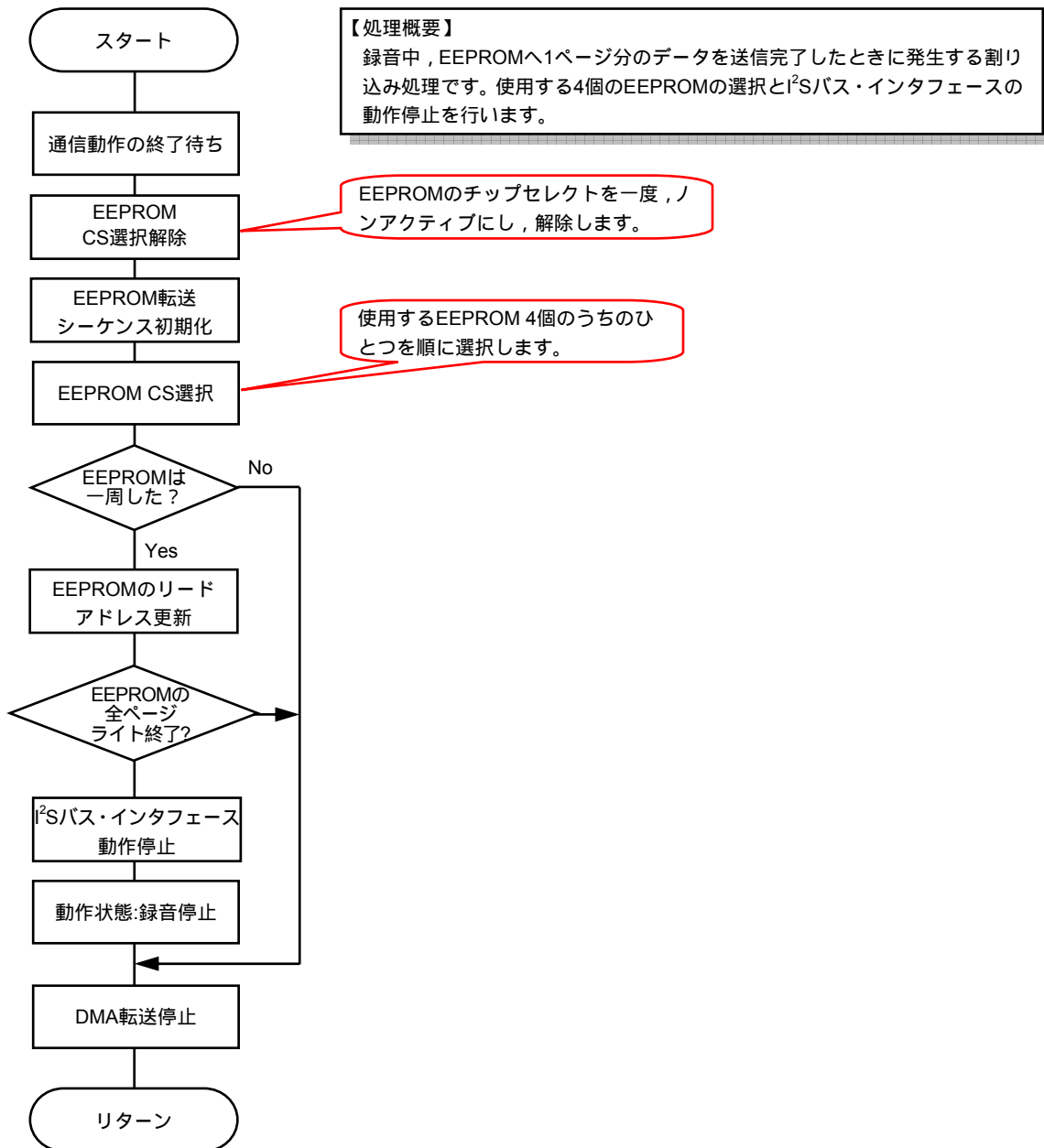
< SPIインタフェース用INTCSI10割り込み処理 >



< INTDMA0 割り込み処理 >



< INTDMA1 割り込み処理 >

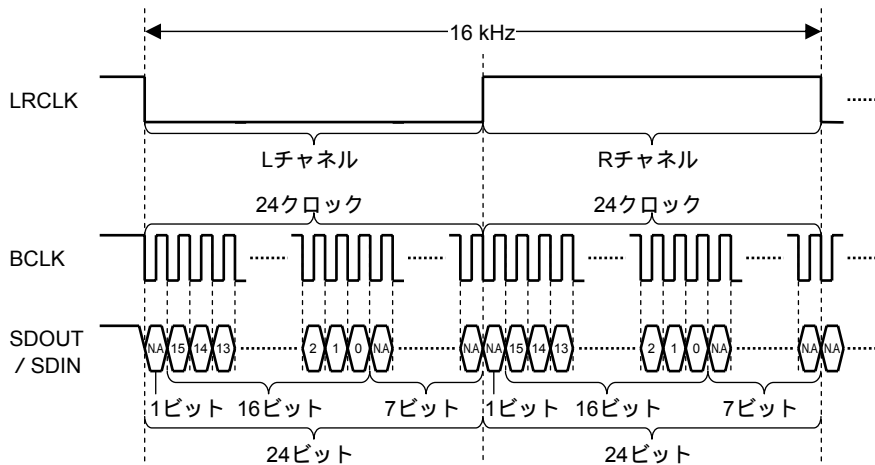


3.5 I²Sバス・インタフェースのフォーマット

I²Sバス・インタフェースのフォーマットを示します。

【オーディオCODECとの送受信波形】

以下にオーディオCODECとの送受信波形を示します。



LRCLK :

オーディオ・データのサンプリング周波数となります。

プログラムでは、タイマ・アレイ・ユニット (TAU) のチャンネル0のインターバル・タイマ・モードを使用し、 16×2 kHz のインターバルでTO00出力を行います。また、タイマ・アレイ・ユニット (TAU) のチャンネル0は、INTTM00割り込みを発生させ、BCLK、SDOUTおよびSDINとの同期が行えるようにします。

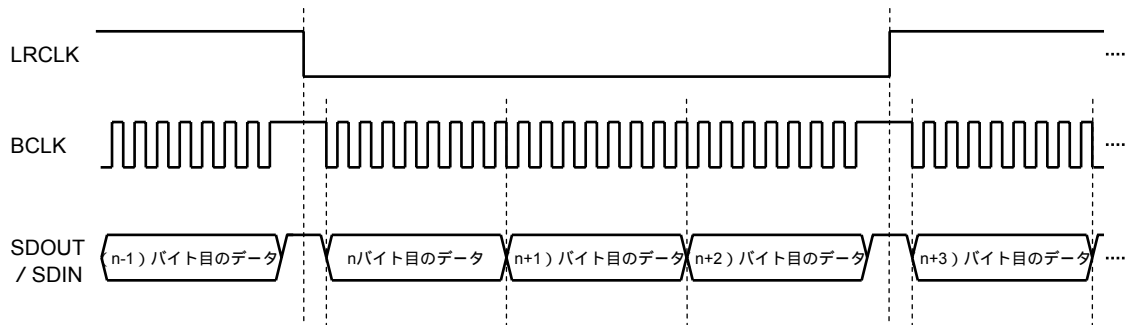
BCLK / SDOUT / SDIN :

オーディオ・データのデータ入出力となります。

プログラムでは、シリアル・アレイ・ユニット0 (SAU0) のチャンネル0のCSI00を使用したマスタ連続送受信を行います。3バイトのデータ転送が、LRCLKの半クロックに収まるように転送速度を833 kHzとしています。また、シリアル・アレイ・ユニット0 (SAU0) のチャンネル0は、バッファ空き割り込み (INTCSI00) を発生させ、連続送受信を行えるようにします。

【プログラムでのタイミング】

プログラムでのタイミングを示します。



INTTM00割り込みのタイミングです。

- ・送信時は、 n バイト目のデータを送受信バッファ・レジスタ (SIO00) (SDR00レジスタの下位8ビット) にライトし、CSI00のマスタ連続送信動作を再開します。
- ・受信時は、送受信バッファ・レジスタ (SIO00) をリードし、 $(n-1)$ バイト目のデータを保存します。ダミー・データを送受信バッファ・レジスタ (SIO00) にセットし、CSI00のマスタ連続受信動作を再開します。

1度目のINTCSI00バッファ空き割り込みのタイミングです。

- ・送信時は、 $n+1$ バイト目のデータを送受信バッファ・レジスタ (SIO00) にライトします。
- ・受信時は、このタイミングでの送受信バッファ・レジスタ (SIO00) には、有効なデータではないのでリードしません。連続受信を継続するため、ダミー・データを送受信バッファ・レジスタ (SIO00) にセットします。

2度目のINTCSI00バッファ空き割り込みのタイミングです。

- ・送信時は、 $n+2$ バイト目のデータを送受信バッファ・レジスタ (SIO00) にライトします。
- ・受信時は、送受信バッファ・レジスタ (SIO00) をリードし、 n バイト目のデータを保存します。連続受信を継続するため、ダミー・データを送受信バッファ・レジスタ (SIO00) にセットします。

3度目のINTCSI00バッファ空き割り込みのタイミングです。

- ・送信時は、送信動作を行わずに の動作再開を待ちます。
- ・受信時は、送受信バッファ・ (SIO00) のリードのみを行い、 $(n+1)$ バイト目のデータを保存します。ダミー・データのセットは行わずに の動作再開を待ちます。

第4章 設定方法について

この章では、I²Sバス・インタフェースを使用するための設定、使用する周辺の初期設定、メイン処理、各割り込み処理、サブルーチンについて説明します。

初期設定方法の詳細については、78K0R/Kx3 サンプル・プログラム（使用する周辺の初期設定） LED点灯のスイッチ制御偏 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、各製品のユーザーズ・マニュアル(78K0R/KE3, 78K0R/KF3, 78K0R/KG3, 78K0R/KH3, 78K0R/KJ3) を参照してください。

アセンブラ命令については、78K0Rマイクロコントローラ 命令編 ユーザーズ・マニュアルを参照してください。

4.1 I²Sバス・インタフェースを使用するための設定

本サンプル・プログラムでは、I²Sバス・インタフェースを接続し、オーディオ・データの送受信ができるようにLRクロック出力用にタイマ・アレイ・ユニット (TAU) のチャンネル0のTO00出力設定、データ部分の送受信用にシリアル・アレイ・ユニット0 (SAU0) のチャンネル0のCSI00の設定を行います。

【タイマ・アレイ・ユニット (TAU) の設定】

(1) タイマ・アレイ・ユニット (TAU) の入力クロックの制御

タイマ・アレイ・ユニット (TAU) を使用できるように、周辺ハードウェア・マクロの使用を可能にするための周辺イネーブル・レジスタ0 (PER0) の設定を行います。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ることができます。

図4-1 周辺イネーブル・レジスタ0 (PER0) のフォーマット

アドレス : F00F0H

RTCEN	DACEN	ADCEN	IIC0EN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
-------	-------	-------	--------	--------	--------	--------	--------

タイマ・アレイ・ユニット (TAU0EN) の入力クロックの制御

0	入力クロック供給停止 ・タイマ・アレイ・ユニットで使用するSFRへのライト不可 ・タイマ・アレイ・ユニットはリセット状態
1	入力クロック供給 ・タイマ・アレイ・ユニットで使用するSFRへのリード/ライト可

- 注意1. タイマ・アレイ・ユニットの設定をする際には、必ず最初にTAU0EN = 1の設定を行ってください。TAU0EN = 0の場合は、タイマ・アレイ・ユニットの制御レジスタへの書き込みは無視され、読み出し値もすべて初期値となります。(タイマ入力選択レジスタ0 (TIS0), 入力切り替え制御レジスタ (ISC), ノイズ・フィルタ許可レジスタ1 (NFEN1), ポート・モード・レジスタ0, 1, 3, 4, 13, 14 (PM0, PM1, PM3, PM4, PM13, PM14), ポート・レジスタ0, 1, 3, 4, 13, 14 (P0, P1, P3, P4, P13, P14) は除く)。
2. TAU1EN は、78K0R/KJ3, 78K0R/KH3 のみに存在します。DACEN は、78K0R/KJ3, 78K0R/KH3, 78K0R/KG3, 78K0R/KF3 のみに存在します。存在しないビットは、必ず0を設定してください。

(2) 動作クロック (CK00) の選択

タイマ・アレイ・ユニット (TAU) のチャンネル0, 1, 2に供給される2種類の動作クロック (CK00, CK01) を選択するため, タイマ・クロック選択レジスタ0 (TPS0) の設定を行います。TPS0のビット7-4でCK01を, ビット3-0でCK00を選択します。

本サンプル・プログラムでは, チャンネル0, 2にて動作クロックCK00を使用し, チャンネル1にて動作クロックCK01を使用しています。I²Sバス・インタフェースのLRクロック用には, チャンネル0にてCK00を使用します。

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

アドレス : F01B6H , F01B7H

0	0	0	0	0	0	0	0	PR	S013	PR	S012	PR	S011	PR	S010	PR	S003	PR	S002	PR	S001	PR	S000
---	---	---	---	---	---	---	---	----	------	----	------	----	------	----	------	----	------	----	------	----	------	----	------

動作クロック (CK00) の選択

0	0	0	0	fCLK
0	0	0	1	fCLK/2
0	0	1	0	fCLK/2 ²
0	0	1	1	fCLK/2 ³
0	1	0	0	fCLK/2 ⁴
0	1	0	1	fCLK/2 ⁵
0	1	1	0	fCLK/2 ⁶
0	1	1	1	fCLK/2 ⁷
1	0	0	0	fCLK/2 ⁸
1	0	0	1	fCLK/2 ⁹
1	0	1	0	fCLK/2 ¹⁰
1	0	1	1	fCLK/2 ¹¹
1	1	1	1	INTTM03
上記以外				設定禁止

動作クロック (CK01) の選択 (本サンプル・プログラムでは未使用)

0	0	0	0	fCLK
0	0	0	1	fCLK/2
0	0	1	0	fCLK/2 ²
0	0	1	1	fCLK/2 ³
0	1	0	0	fCLK/2 ⁴
0	1	0	1	fCLK/2 ⁵
0	1	1	0	fCLK/2 ⁶
0	1	1	1	fCLK/2 ⁷
1	0	0	0	fCLK/2 ⁸
1	0	0	1	fCLK/2 ⁹
1	0	1	0	fCLK/2 ¹⁰
1	0	1	1	fCLK/2 ¹¹
1	1	1	1	INTTM03
上記以外				設定禁止

注意 ビット15-8には, 必ず0を設定してください。

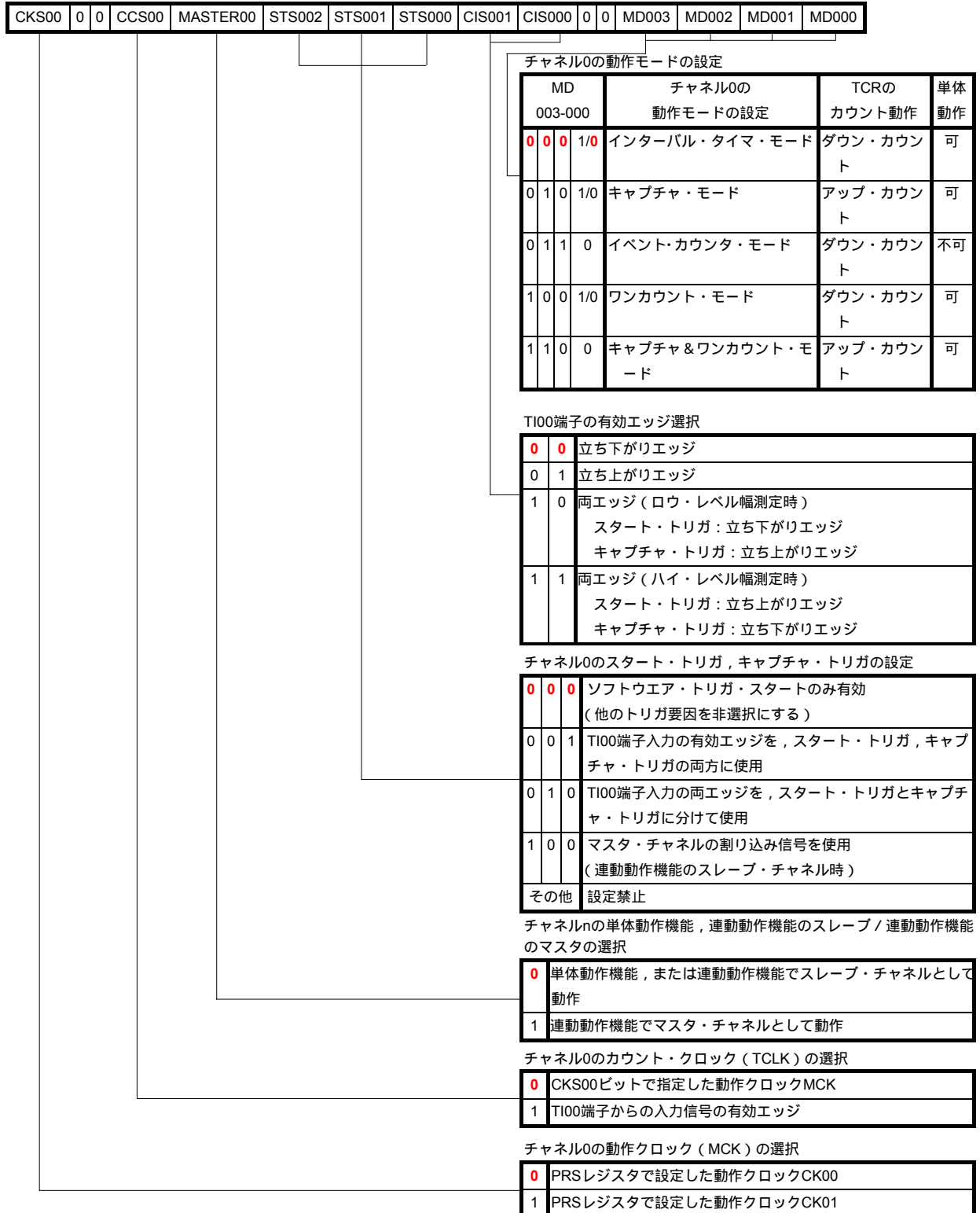
備考 fCLK : CPU / 周辺ハードウェア・クロック周波数

(3) 動作モードなどの設定

I²Sバス・インタフェースのLRクロック用に、タイマ・アレイ・ユニット (TAU) で使用する動作クロック (MCK) の選択, タイマ・クロック (TCLK) 入力の使用可否, タイマ入力端子の有効エッジの設定, 動作モード設定をタイマ・モード・レジスタ00 (TMR00) にて行います。

図4-3 タイマ・モード・レジスタ00 (TMR00) のフォーマット

アドレス: F0190H, F0191H



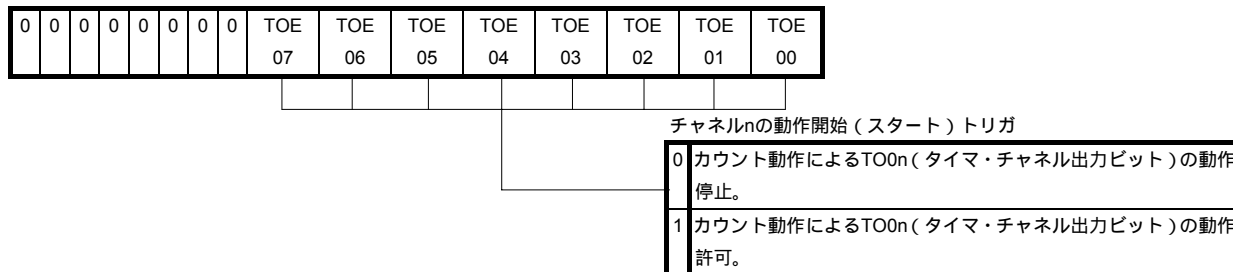
注意 ビット14, 13, 5, 4には, 必ず0を設定してください。

(4) タイマ出力の許可

タイマ出力の許可はチャンネルごとに設定するタイマ出力許可レジスタ0 (TOE0) で行います。I²Sバス・インタフェースのLRクロック用には、チャンネル0のTOE00ビットにて設定します。

図4 - 4 タイマ出力許可レジスタ0 (TOE0) のフォーマット

アドレス : F01BAH, F01BBH



- 注意1. TOE0のビット15-8には、必ず0を設定してください。
2. TOE07は、78K0R/KJ3, 78K0R/KH3, 78K0R/KG3, 78K0R/KF3のみに存在します。存在しないビットは、必ず0を設定してください。

備考 n : チャンネル番号 (n = 0-7)

(5) カウント動作開始の許可

タイマのカウント動作開始の許可はチャンネルごとに設定するタイマ・チャンネル開始レジスタ0 (TS0) で行います。タイマ・チャンネル開始レジスタ0 (TS0) は、トリガ・レジスタですので、タイマのカウント動作が許可状態になるとクリアされます。

I²Sバス・インタフェースのLRクロックの動作開始時には、チャンネル0のTS00ビットを1にします。

図4 - 5 タイマ・チャンネル開始レジスタ0 (TS0) のフォーマット

アドレス : F01B2H, F01B3H



注意 TS0のビット15-8には、必ず0を設定してください。

備考 n : チャンネル番号 (n = 0-3)

(6) カウント動作停止の許可

タイマのカウント動作停止は、チャンネルごとに設定するタイマ・チャンネル停止レジスタ0 (TT0) で行います。タイマ・チャンネル停止レジスタ0 (TT0) は、トリガ・レジスタですので、タイマのカウント動作が停止状態になるとクリアされます。

I²Sバス・インタフェースのLRクロックの動作停止時には、チャンネル0のTT00ビットを1にします。

図4-6 タイマ・チャンネル停止レジスタ0 (TT0) のフォーマット

アドレス : F01B4H, F01B5H



【シリアル・アレイ・ユニット0 (SAU0) の設定】

(1) シリアル・アレイ・ユニット0 (SAU0) の入力クロックの制御

シリアル・アレイ・ユニット0 (SAU0) を使用できるように、周辺ハードウェア・マクロの使用を可能にするための周辺イネーブル・レジスタ0 (PER0) の設定を行います。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ることができます。

図4 - 7 周辺イネーブル・レジスタ0 (PER0) のフォーマット

アドレス : F00F0H

RTCEN	DACEN	ADCEN	IIC0EN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
-------	-------	-------	--------	--------	---------------	--------	--------

シリアル・アレイ・ユニット0 (SAU0EN) の入力クロックの制御	
0	入力クロック供給停止 ・シリアル・アレイ・ユニット0で使用するSFRへのライト不可 ・シリアル・アレイ・ユニット0リセット状態
1	入力クロック供給 ・シリアル・アレイ・ユニット0で使用するSFRへのリード/ライト可

- 注意1. シリアル・アレイ・ユニットmの設定をする際には、必ず最初にSAUmEN = 1の設定を行ってください。SAUmEN = 0の場合は、シリアル・アレイ・ユニットmの制御レジスタへの書き込みは無視され、読み出しても値はすべて初期値となります（入力切り替え制御レジスタ (ISC)、ノイズフィルタ許可レジスタ (NFEN0)、ポート入力モード・レジスタ (PIM0)、ポート出力モード・レジスタ (POM0)、ポート・モード・レジスタ (PM0, PM1)、ポート・レジスタ (P0, P1) は除く)。
- PER0レジスタを“1”に設定後に、4クロック以上間隔をあけてからSPSmレジスタを設定してください。
 - TAU1ENは、78K0R/KJ3, 78K0R/KH3のみに存在します。DACENは、78K0R/KJ3, 78K0R/KH3, 78K0R/KG3, 78K0R/KF3のみに存在します。存在しないビットは、必ず0を設定してください。

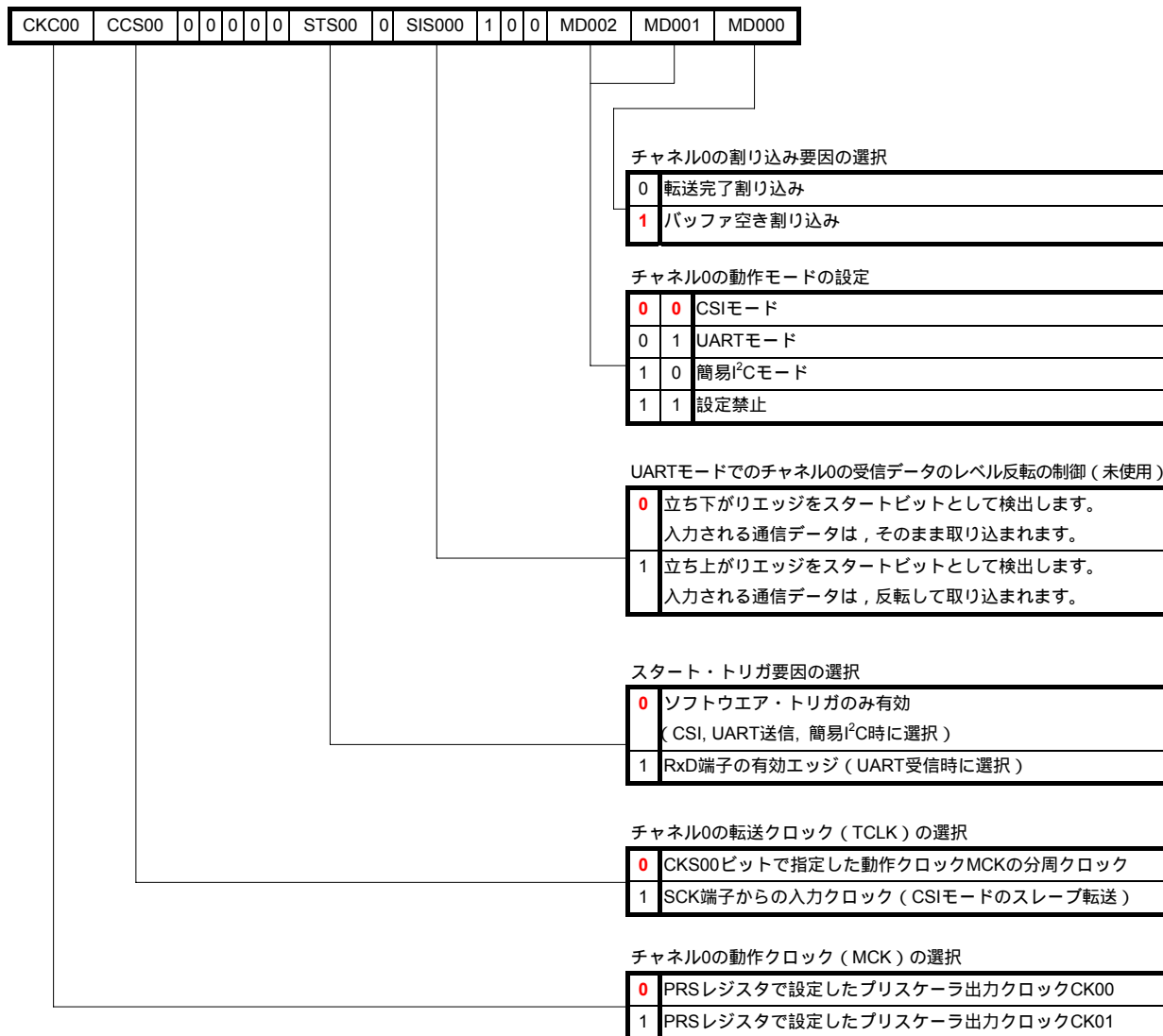
備考 m : ユニット番号 (m = 0, 1)

(3) 動作モードなどの設定

I²Sバス・インタフェースのデータ送受信用に、シリアル・アレイ・ユニット0 (SAU0) のCSI00で使用する動作クロック (MCK) の選択, シリアル・クロック (SCK) 入力の使用可否, スタート・トリガ設定, 動作モード (CSI, UART, I²C) 設定, 割り込み要因の選択をシリアル・モード・レジスタ00 (SMR00) に行います。

図4 - 9 シリアル・モード・レジスタ00 (SMR00) のフォーマット

アドレス : F0110H, F0111H



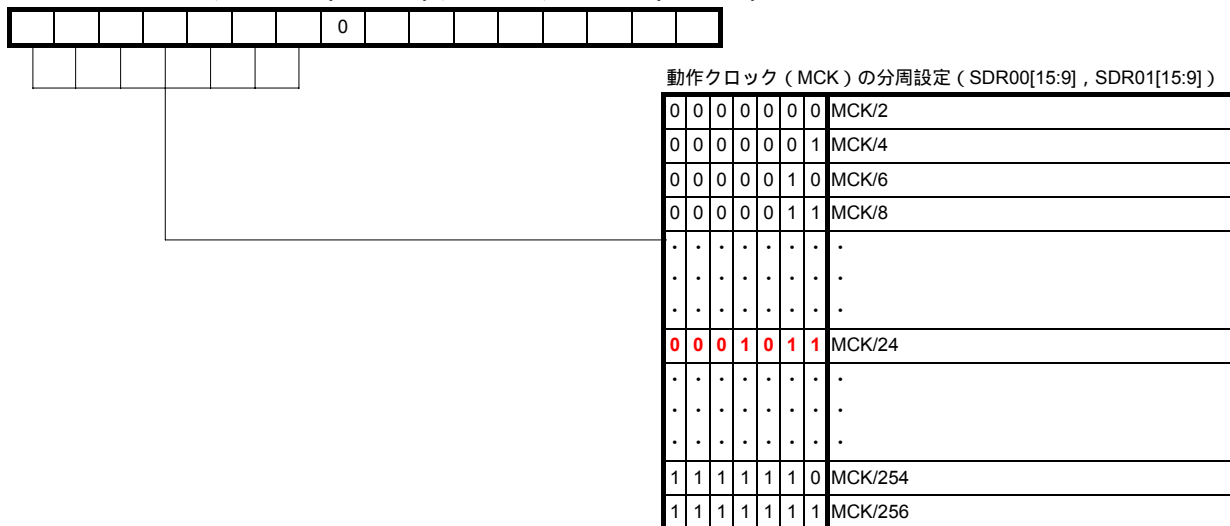
注意 ビット13-9, 7, 4, 3には、必ず0を設定してください。ビット5には、必ず1を設定してください。

(4) 動作クロックの分周設定

シリアル・アレイ・ユニット0 (SAU0) のCSI00で使用する動作クロック (MCK) の分周設定は、シリアル・データ・レジスタ00 (SDR00) の16ビットのレジスタの上位7ビットで行います。なお、シリアル・データ・レジスタ00 (SDR00) の下位8ビットは送受信データ・レジスタ (SIO00) となります。

図4 - 10 シリアル・データ・レジスタ00 (SDR00) のフォーマット

アドレス : FFF10H, FFF11H (SDR00) , FFF12H, FFF13H (SDR01)



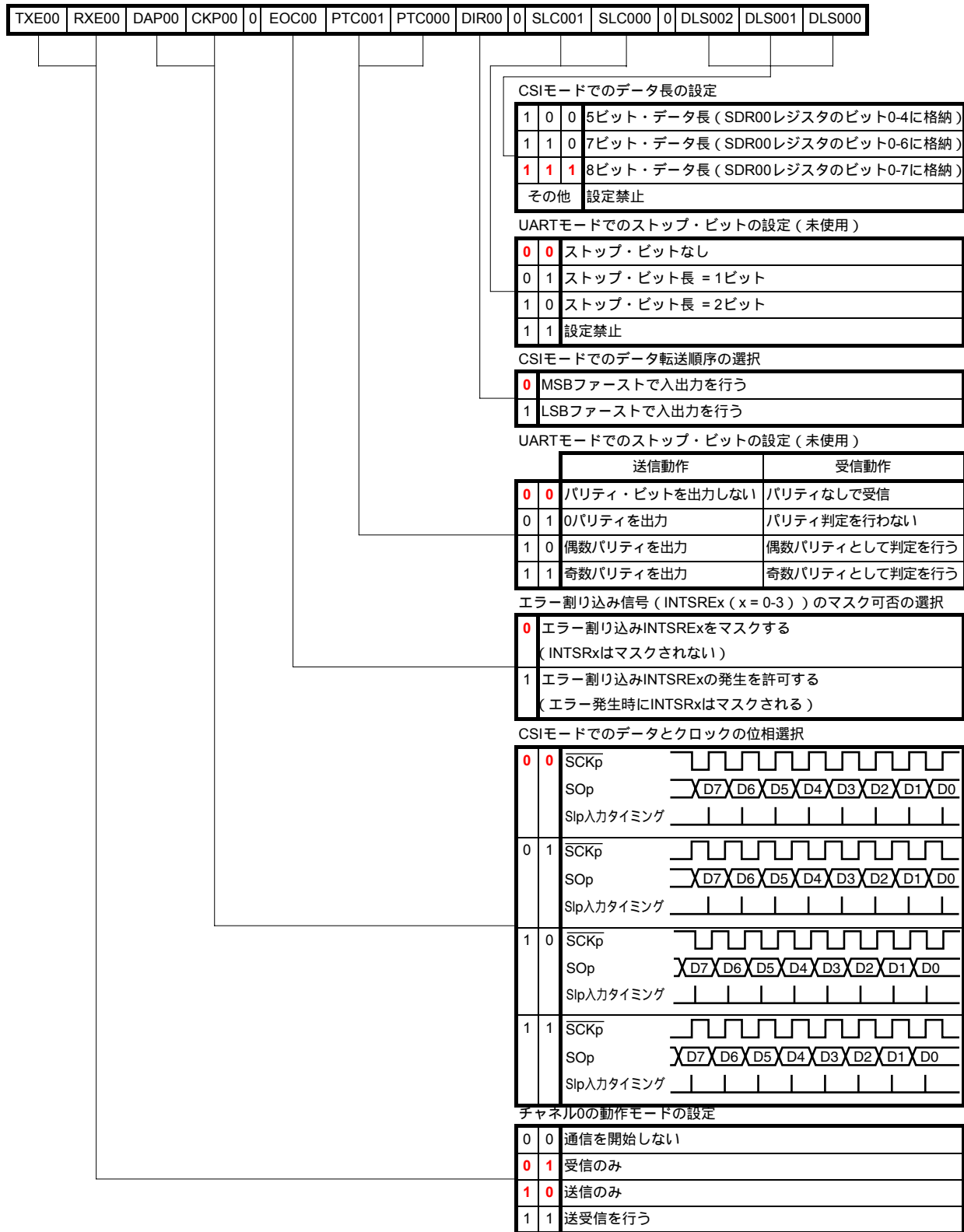
注意 ビット8は、必ず0を設定してください。

(5) 通信動作の設定

通信動作の設定は、シリアル通信動作設定レジスタ00 (SCR00) にて行います。データ送受信モード、データとクロックの位相、データ長、などの設定を行います。

図4 - 11 シリアル通信動作設定レジスタ00 (SCR00) のフォーマット

アドレス : F0118H, F0119H



注意 ビット11, 6, 3には、必ず0を設定してください。ビット2には、必ず1を設定してください。

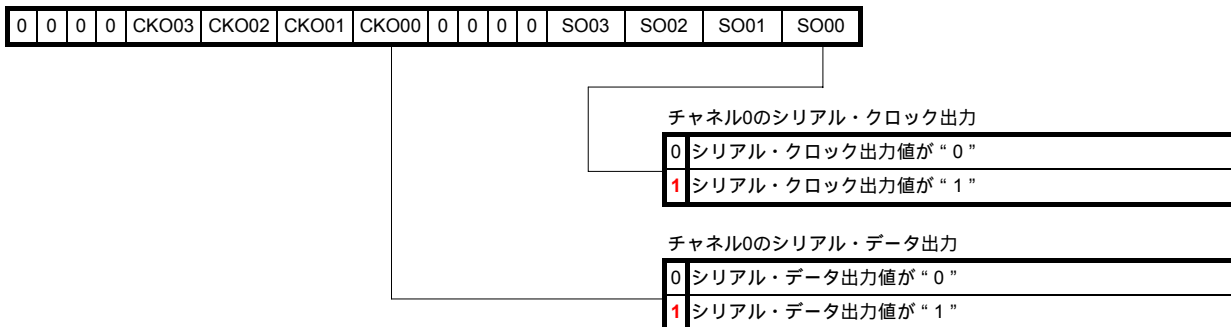
(6) 初期出力レベルの設定

シリアル出力レジスタ0 (SO0) にて, シリアル・アレイ・ユニット0 (SAU0) のCSI送受信での初期出力レベルの設定を行います。

I²Sバス・インタフェースのデータ送信用には, 使用する周辺の初期設定にてチャンネル0のCK00ビットを1, SO00ビットを1にします。

図4 - 12 シリアル出力レジスタ0 (SO0) のフォーマット

アドレス : F0128H, F0129H



注意 SO0のビット15-12, 7-4には, 必ず0を設定してください。CKO03, SO03は, 78K0R/KJ3, 78K0R/KH3マイクロコントローラにのみ存在します。CKO01, SO01は, 78K0R/KJ3, 78K0R/KH3, 78K0R/KG3, 78K0R/KF3マイクロコントローラにのみ存在します。存在しないCKO01, CKO03, SO01, SO03は, 必ず1を設定してください。

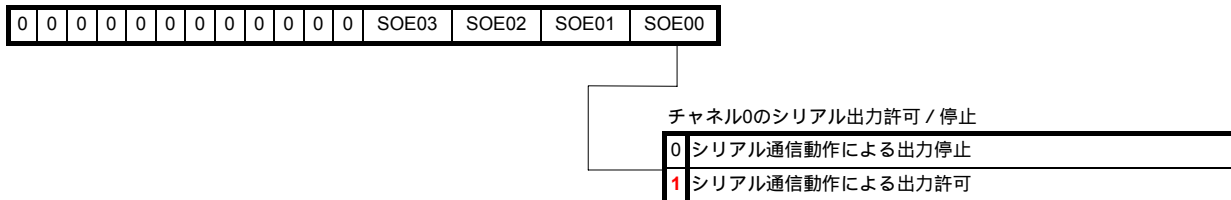
(7) シリアル通信動作の出力許可 / 停止の設定

シリアル・アレイ・ユニット0 (SAU0) のCSI00で使用するシリアル通信動作の出力許可 / 停止は, シリアル出力許可レジスタ0 (SOE0) の設定で行います。シリアル出力を許可したチャンネルは, 通信動作によって反映された値がシリアル・データ出力端子から出力されます。

I²Sバス・インタフェースのデータ送信用には, 使用する周辺の初期設定にてCSI00送信動作の出力許可時にチャンネル0のSOE00ビットを1にします。

図4 - 13 シリアル出力許可レジスタ0 (SOE0) のフォーマット

アドレス : F012AH, F012BH (SOE0)



注意 SOE0のビット15-4には, 必ず0を設定してください。SOE03は, 78K0R/KJ3, 78K0R/KH3マイクロコントローラにのみ存在します。SOE01は, 78K0R/KJ3, 78K0R/KH3, 78K0R/KG3, 78K0R/KF3マイクロコントローラにのみ存在します。存在しないビットは, 必ず0を設定してください。

(8) シリアル通信 / カウント開始の許可

シリアル・アレイ・ユニット0 (SAU0) のCSI00で使用するシリアルの通信 / カウント開始の許可はチャンネルごとに設定するシリアル・チャンネル開始レジスタ0 (SS0) で行います。シリアル・チャンネル開始レジスタ0 (SS0) は、トリガ・レジスタですので、シリアル送受信の動作が許可状態になるとクリアされます。

I²Sバス・インタフェースのデータ送受信開始時には、チャンネル0 (CSI00) のSS00ビットを1にします。

図4 - 14 シリアル・チャンネル開始レジスタ0 (SS0) のフォーマット

アドレス : F0122H, F0123H (SS0)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	SS03	SS02	SS01	SS00
---	---	---	---	---	---	---	---	---	---	---	---	---	---	------	------	------	------

チャンネル0の動作開始トリガ

0	トリガ動作せず
1	通信待機状態に遷移する (すでに通信動作中の場合は、通信動作を停止し、起動条件待ちの状態となる)

注意 SS0のビット15-4には、必ず0を設定してください。

(9) シリアル通信 / カウント停止の許可

通信 / カウント停止の許可はチャンネルごとに設定するシリアル・チャンネル停止レジスタ0 (ST0) で行います。シリアル・チャンネル停止レジスタ0 (ST0) は、トリガ・レジスタですので、シリアル送受信の動作が停止状態になるとクリアされます。

I²Sバス・インタフェースのデータ送受信停止時には、チャンネル0 (CSI00) のST00ビットを1にします。

図4 - 15 シリアル・チャンネル停止レジスタ0 (ST0) のフォーマット

アドレス : F0124H, F0125H (ST0)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	ST03	ST02	ST01	ST00
---	---	---	---	---	---	---	---	---	---	---	---	---	---	------	------	------	------

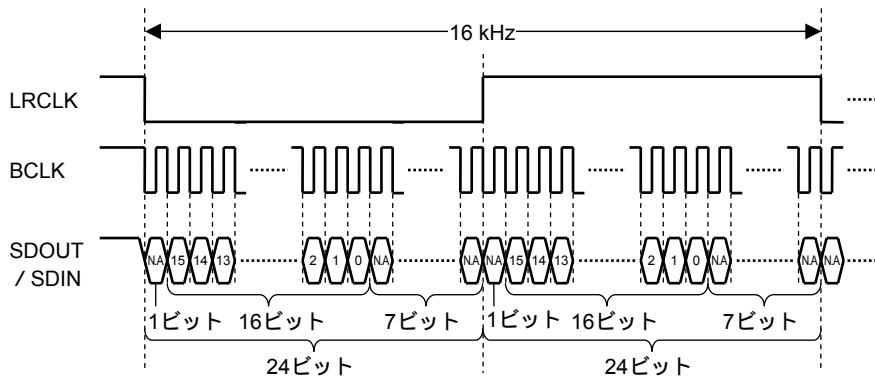
チャンネル0の動作停止トリガ

0	トリガ動作せず
1	通信動作を停止する (制御レジスタ, シフト・レジスタの値, およびシリアル・クロック入出力端子, シリアル・データ出力端子, FEF, PEF, OVFの各エラー・フラグの状態を保持したまま停止)

注意 ST0のビット15-4には、必ず0を設定してください。

【例】 下図に示されるような²Sバス・インタフェースに対応させる。(サンプル・プログラムと同内容)

LRCLKをタイマ・アレイ・ユニット(TAU)のチャンネル0のインターバル・タイマ・モードを使用し, 16×2 kHzのインターバルでT000出力を行い, BCLK, SDOUTおよびSDINのオーディオ・データのデータ入出力には, シリアル・アレイ・ユニット0 (SAU0) のチャンネル0のCSI00を使用したマスタ連続送受信を行います。



設定手順

- タイマ・アレイ・ユニット0用のポート出力ラッチの初期設定 (P01を1)
- タイマ・アレイ・ユニット0用のポート・モードの設定 (PM01を0)
- タイマ・アレイ・ユニット0の入力クロック供給 (TAU0ENを1)
- タイマ・アレイ・ユニット0の動作クロック (TPS0) の選択
- タイマ・アレイ・ユニット0チャンネル0の動作モード (TMR00) の設定
- タイマ・アレイ・ユニット0チャンネル0のタイマ・カウンタ (TDR00) の設定
- シリアル・アレイ・ユニット0の入力クロック供給 (SAU0ENを1)
- シリアル・アレイ・ユニット0のチャンネル0の動作クロック (SPS0) の選択
- シリアル・アレイ・ユニット0のチャンネル0の転送クロック (SDR00) の設定
- シリアル・アレイ・ユニット0のチャンネル0の動作モード (SMR00) の設定
- シリアル・アレイ・ユニット0のチャンネル0の初期データ出力 (SO00) の設定
- シリアル・アレイ・ユニット0のチャンネル0の出力許可 (SOE0)
- シリアル・アレイ・ユニット0のチャンネル0用のポート出力ラッチの初期設定 (P10, P12)
- シリアル・アレイ・ユニット0のチャンネル0用のポート・モードの設定 (PM10, PM11, PM12)
- シリアル・アレイ・ユニット0のチャンネル0の動作開始 (SS00を1)
- INTCSI00に対応する割り込み要求フラグ (CSIF00を0) をクリア
- INTCSI00に対応する割り込みマスク・フラグ (CSIMK00) をクリア
- EI命令を実行し, 割り込みを許可
- シリアル・アレイ・ユニット0のシリアル・データ・レジスタ (SDR00) の設定
- シリアル・アレイ・ユニット0のチャンネル0通信フォーマット (SCR00) の設定
- ⑳タイマ・アレイ・ユニット0のチャンネル0タイマ出力の許可 (TOE00を1)
- ㉑INTTM00に対応する割り込み要求フラグ (PIF0を0) をクリア
- ㉒INTTM00に対応する割り込みマスク・フラグ (PMK0) をクリア
- ㉓シリアル・アレイ・ユニット0のチャンネル0のCSI00動作開始 (SS00を1)
- ㉔タイマ・アレイ・ユニット0用のチャンネル0ポート出力ラッチの初期出力設定 (P01を0)
- ㉕タイマ・アレイ・ユニット0のチャンネル0チャンネル0の動作開始 (TS00を1)
- ㉖シリアル・アレイ・ユニット0のチャンネル0のSIO00にダミー・データ・セット

上述の ~㉔と, 次ページの ~㉖は対応しています。

C言語のプログラム例 (サンプル・プログラムと同内容)

```

... (略) (プログラムの処理前) ...
#pragma SFR /* 特殊機能レジスタ(SFR)名を記述可能にする */
#pragma EI /* EI命令を記述可能にする */

... (略) (プログラムの処理内) ...

/* -----
   ポート0の設定
   ----- */
P0 = 0b00011010; /* P00,P02,P05-P06の出力ラッチLow */
PM0 = 0b10000000; /* P01,P03-P04の出力ラッチHIGH */
/* P00-P06を出力ポートに設定 */

... (略) ...

/* タイマ・アレイユニットのタイマ・クロック選択 */
TAU0EN = 1; /* タイマ・アレイユニットの入力クロック供給 */
TPS0L = 0b10010000; /* 動作クロック選択 */
/* |||++++----- CK00: fCLK */
/* ++++----- CK01: fCLK/2^9 */

... (略) ...

TMR00 = 0b0000000000000000; /* 動作モード設定 */
/* |||++++----- MD003-000: インターバル・タイマ・モード */
/* +++----- <00固定> */
/* +++----- CIS001-000: 未使用 */
/* +++----- STS002-000: ソフトウェア・トリガ・スタートのみ有効 */
/* +++----- MASTER00: 単体動作 */
/* +++----- CSS00: CKS00ビットで指定したマクロ・クロックMCK */
/* +++----- <00固定> */
/* +++----- CKS00: PRSレジスタで設定した動作クロックCK00 */
TDR00 = 625-1; /* インターバル設定: 16kHz出力(32kHz) */

... (略) ...

SAU0EN = 1; /* シリアル・アレイ・ユニットの入力クロック供給 */

NOP(); /* ウェイト */
NOP();
NOP();
NOP();

SPS0L = 0b00000000; /* 動作クロック選択 fCLK */
/* |||++++----- PRS003-00: fCLK */
/* +++----- PRS013-10: 未使用 */

SDR00 = (12-1) << 9; /* ビット15-7: 転送クロック設定(833kHz) */

SMR00 = 0b0000000000100001; /* 動作モード選択: CSIモード */
/* |||++++----- MD000: バッファ空き割り込み */
/* +++----- MD002-1: CSIモード */
/* +++----- <100固定> */
/* +++----- SIS000: 未使用 */
/* +++----- <0固定> */
/* +++----- STS00: ソフトウェア・トリガのみ有効(CSI時は固定) */
/* +++----- <00000固定> */
/* +++----- CSS00: 転送クロックをCKS00で指定した動作クロックMCK */
/* +++----- CKS00: 動作クロックをPRSで設定したに設定 */

... (略) ...

/* 初期データ出力(CSI10の設定時に同時に行う) */
SO0 = 0b00000000100000001; /* SO端子&SCK端子の初期出力設定 */
/* |||++++----- SO0n-0: 送信で使用 */
/* +++----- CK0n-0: チャンネルn-0のシリアル・クロック出力 */

... (略) ...

SOE0L.0 = 1; /* 出力許可 */

P_SCK00 = 1; /* SCK00ラッチ:ハイ */
PM_SCK00 = 0; /* SCK00端子出力設定 */
PM_SI00 = 1; /* SI00端子入力設定 */
P_SO00 = 1; /* SO00ラッチ:ハイ */
PM_SO00 = 0; /* SO00端子出力設定 */

SS0L.0 = 1; /* CSI00動作開始(トリガ・ビット) */
CSIF00 = 0; /* INTCSI00割り込み要求クリア */
CSIMK00 = 0; /* INTCSI00割り込み処理許可 */

... (略) ...

EI(); /* 割り込み許可 */

... (略) ...

```

特殊機能レジスタ(SFR)名を記述可能にする#pragma指令です。

割り込み許可(EI)命令を記述可能にする#pragma指令です。

4.2 変数と定数の定義

アセンブリ言語で使用する変数と定数の定義です。

ポート定義

ポート定義は、プログラム内で使用するポートを明確にするためと、ポートとポート・モードの設定が何の機能のためのものかをわかりやすくするために使用しています。

```

;
;
;      ポート定義
;
;=====
;オーディオCODEC リセット
P_RESETB      EQU      P7.3          ;リセット
PM_RESETB     EQU      PM7.3        ;リセット出力

; オーディオCODEC内のレジスタ設定用I2Cインタフェース
P_SCL         EQU      P6.0          ;SCL0ラッチ
PM_SCL        EQU      PM6.0        ;SCL0出力
P_SDA         EQU      P6.1          ;SDA0ラッチ
PM_SDA        EQU      PM6.1        ;SDA0入出力

;オーディオCODEC I2Sバス・インタフェース
P_LRCLK       EQU      P0.1          ;LRCLK
PM_LRCLK      EQU      PM0.1        ;LRCLK出力
P_SCK00       EQU      P1.0          ;SCK00
PM_SCK00      EQU      PM1.0        ;SCK00出力
PM_SI00       EQU      PM1.1        ;SI00入力
P_SO00        EQU      P1.2          ;SO00
PM_SO00       EQU      PM1.2        ;SO00出力

;キー入力(2本使用)
P_KEY         EQU      P7            ;KR(P70-P71を使用)
PM_KEY        EQU      PM7          ;KR入力(PM70-PM71を使用)
P_PLAYKY      EQU      P7.0         ;再生キー
PM_PLAYKY     EQU      PM7.0        ;再生キー入力
P_RECKY       EQU      P7.1         ;録音キー
PM_RECKY      EQU      PM7.1        ;録音キー入力

;動作中インジケータ用LED出力(1本使用)
P_LED         EQU      P7.2          ;動作中インジケータ(Lowアクティブ)
PM_LED        EQU      PM7.2        ;動作中インジケータ(LED)出力
;動作中に点灯させ、キー入力が可能なときに消灯させます。

;EEPROM SPIインタフェース(4個使用)
P_CS          EQU      P7            ;チップ・セレクトCS(P73-P77を使用)
PM_CS         EQU      PM7          ;チップ・セレクトCS出力(PM73-PM77を使用)
P_CS3         EQU      P7.7         ;CS3
PM_CS3        EQU      PM7.7        ;CS3出力
P_CS2         EQU      P7.6         ;CS2
PM_CS2        EQU      PM7.6        ;CS2出力
P_CS1         EQU      P7.5         ;CS1
PM_CS1        EQU      PM7.5        ;CS1出力
P_CS0         EQU      P7.4         ;CS0
PM_CS0        EQU      PM7.4        ;CS0出力
P_SCK10       EQU      P0.4          ;SCK10
PM_SCK10      EQU      PM0.4        ;SCK10出力
PM_SI10       EQU      PM0.3        ;SI10入力
P_SO10        EQU      P0.2         ;SO10
PM_SO10       EQU      PM0.2        ;SO10出力

```

変数と定数の定義

主に、全体の動作に関わるもの、オーディオCODECとのI²Sバス・インタフェースの動作に関わるもの、EEPROMとのSPI動作に関わるもの、録音/再生時にデータを一時的に保存するための内蔵RAM内のメモリ領域に大別されます。

```

;
;
;          RAMの定義
;
;=====
DPLAY    DSEG    SADDR
;全体の動作
RPLAYMOD:      DS      1      ;動作状態
CRESET        EQU     0      ;リセット
CSTOP         EQU     1      ;停止中
CREC          EQU     2      ;録音中
CREC_END      EQU     3      ;録音終了中
CPLAY_START   EQU     4      ;再生準備開始
CPLAY_SET     EQU     5      ;再生準備中
CPLAY         EQU     6      ;再生中
CPLAY_END     EQU     7      ;再生終了中

;オーディオCODECとのI2Sバス・インタフェース動作関連
RCSI00CNT:    DS      1      ;I2Sデータの1バイト送受信カウンタ
RPLAYINFO:    DS      1      ;プレイ情報
FI2SPAGE      EQU     RPLAYINFO.7 ; I2Sバスの送受信中のページ: ページ1(0) / ページ2(1)

;EEPROMとのSPIインタフェース動作関連
RP_CS:        DS      1      ;EEPROMのチップ・セレクトCs設定値
REEPSEQ:      DS      1      ;EEPROMの転送シーケンス
CEEPSEQ_RESET EQU     0      ;リセット状態
CEEPSEQ_WREN  EQU     1      ;ライトイネーブル送信
CEEPSEQ_INST  EQU     2      ;命令バイト送信
CEEPSEQ_ADDRH EQU     3      ;24ビットアドレス上位8ビット部分送信
CEEPSEQ_ADDRM EQU     4      ;24ビットアドレス中位8ビット部分送信
CEEPSEQ_ADDRL EQU     5      ;24ビットアドレス下位8ビット部分送信
CEEPSEQ_DATA  EQU     6      ;DMAによるデータ送信/受信

DPLAYP      DSEG    SADDRP
;オーディオCODECとのI2Sバス・インタフェース動作関連
RI2SADDR:    DS      2      ;I2S送受信データの保存先アドレス

;EEPROMとのSPIインタフェース動作関連
REEPADDR:    DS      2      ;EEPROMのリード/ライト・アドレス(REEPADDR*100H)

DMEM        DSEG    UNITP
;内蔵RAMのメモリ領域
RRECMEM1:    DS      3*2*42  ;ページ1
; LRCLKの42クロック分
; 最終アドレス+1
RRECMEM1E:
RRECMEM2:    DS      3*2*42  ;ページ2
; LRCLKの42クロック分
; 最終アドレス+1
RRECMEM2E:

```

C言語で使用する変数と定数の定義です。

ポート定義

アセンブリ言語と同様に、ポート定義は、プログラム内で使用するポートを明確にするためと、ポートとポート・モードの設定が何の機能のためのものかをわかりやすくするために使用しています。

```

/*=====
                ポート定義
=====*/
/* オーディオCODEC リセット */
#define P_RESETB      P7.3      /* リセット */
#define PM_RESETB     PM7.3     /* リセット出力 */

/* オーディオCODEC内のレジスタ設定用I2Cインタフェース */
#define P_SCL         P6.0      /* SCL0ラッチ */
#define PM_SCL        PM6.0     /* SCL0出力 */
#define P_SDA         P6.1      /* SDA0ラッチ */
#define PM_SDA        PM6.1     /* SDA0入出力 */

/* オーディオCODEC I2Sバス・インタフェース */
#define P_LRCLK       P0.1      /* LRCLK */
#define PM_LRCLK      PM0.1     /* LRCLK出力 */
#define P_SCK00       P1.0      /* SCK00 */
#define PM_SCK00      PM1.0     /* SCK00出力 */
#define PM_SI00       PM1.1     /* SI00入力 */
#define P_SO00        P1.2      /* SO00 */
#define PM_SO00       PM1.2     /* SO00出力 */

/* キー入力 (2本使用) */
#define P_KEY         P7        /* KR(P70-P71を使用) */
#define PM_KEY        PM7       /* KR入力(PM70-PM71を使用) */
#define P_PLAYKY      P7.0      /* 再生キー */
#define PM_PLAYKY     PM7.0     /* 再生キー入力 */
#define P_RECKY       P7.1      /* 録音キー */
#define PM_RECKY      PM7.1     /* 録音キー入力 */

/* 動作中インジケータ用LED出力 (1本使用) */
#define P_LED         P7.2      /* 動作中インジケータ (LOWアクティブ) */
#define PM_LED        PM7.2     /* 動作中インジケータ (LED) 出力 */
/* 動作中に点灯させ、キー入力が可能なときに消灯させます。 */

/* EEPROM SPIインタフェース (4個使用) */
#define P_CS          P7        /* チップ・セレクトCS (P73-P77を使用) */
#define PM_CS         PM7       /* チップ・セレクトCS出力 (PM73-PM77を使用) */
#define P_CS3         P7.7      /* CS3 */
#define PM_CS3        PM7.7     /* CS3出力 */
#define P_CS2         P7.6      /* CS2 */
#define PM_CS2        PM7.6     /* CS2出力 */
#define P_CS1         P7.5      /* CS1 */
#define PM_CS1        PM7.5     /* CS1出力 */
#define P_CS0         P7.4      /* CS0 */
#define PM_CS0        PM7.4     /* CS0出力 */
#define P_SCK10       P0.4      /* SCK10 */
#define PM_SCK10      PM0.4     /* SCK10出力 */
#define PM_SI10       PM0.3     /* SI10入力 */
#define P_SO10        P0.2      /* SO10 */
#define PM_SO10       PM0.2     /* SO10出力

```

変数と定数の定義

アセンブリ言語と同様に、全体の動作に関わるもの、オーディオCODECとのI²Sバス・インタフェースの動作に関わるもの、EEPROMとのSPI動作に関わるもの、録音/再生時にデータを一時的に保存するための内蔵RAM内のメモリ領域に大別されます。

```

/*=====
RAMの定義
=====*/
/* 全体の動作 */
unsigned char ucPlayMode;          /* 動作状態 */
#define CRESET 0                  /* リセット */
#define CSTOP 1                  /* 停止中 */
#define CREC 2                   /* 録音中 */
#define CREC_END 3              /* 録音終了中 */
#define CPLAY_START 4          /* 再生準備開始 */
#define CPLAY_SET 5            /* 再生準備中 */
#define CPLAY 6                /* 再生中 */
#define CPLAY_END 7           /* 再生終了中

/* オーディオCODECとのI2Sバス・インタフェース動作関連 */
unsigned char *ucI2sAddress;      /* I2S送受信データの保存先アドレス */
unsigned char ucI2sByteCounter;  /* I2Sデータの1バイト送受信カウンタ */
boolean bI2sMemoryPage;         /* I2Sバスの送受信中のページ: ページ1(0) / ページ2(1) */

/* EEPROMとのSPIインタフェース動作関連 */
unsigned short ushEepromAddress; /* EEPROMのリード/ライト・アドレス(ushEepromAddress*0x100) */
unsigned char ucEepromCs;        /* EEPROMのチップ・セレクトCs設定値 */
unsigned char ucEepromSeq;       /* EEPROMの転送シーケンス */
#define CEEPSEQ_RESET 0          /* リセット状態 */
#define CEEPSEQ_WREN 1          /* ライトイネーブル送信 */
#define CEEPSEQ_INST 2          /* 命令バイト送信 */
#define CEEPSEQ_ADDRH 3         /* 24ビットアドレス上位8ビット部分送信 */
#define CEEPSEQ_ADDRM 4         /* 24ビットアドレス中位8ビット部分送信 */
#define CEEPSEQ_ADDRL 5         /* 24ビットアドレス下位8ビット部分送信 */
#define CEEPSEQ_DATA 6          /* DMAによるデータ送信/受信

/* 内蔵RAMのメモリ領域 */
unsigned char ucMemoryPage1[3*2*42]; /* ページ1 (LRCLKの42クロック分) */
unsigned char ucMemoryPage2[3*2*42]; /* ページ2 (LRCLKの42クロック分) */

```

4.3 使用する周辺の初期設定

アセンブリ言語の使用する周辺の初期設定では、次の動作を行います。

割り込みを禁止します。

レジスタバンクの設定を行います。

スタック・ポインタの設定を行います。

CPU / 周辺ハードウェア・クロック周波数をX1発振クロック (20 MHz) で動作するように設定を行います。

ポートの設定を行います。

オーディオCODEC内のレジスタ設定を行います。

- (a) オーディオCODECへのクロック供給用にタイマ・アレイ・ユニット (TAU) のチャンネル2のTO02出力の設定を行います。
- (b) プログラム内で使用するウエイト用にタイマ・アレイ・ユニット (TAU) のチャンネル1のインターバル・タイマ・モードの設定を行います。
- (c) オーディオCODECの電源安定待ちとリセットを行います。
- (d) シリアル・インタフェースIIC0を使用し、オーディオCODEC内のレジスタ設定が行えるように設定します。
- (e) オーディオCODEC内のレジスタ設定でシステムONを行います。

プログラム内で使用する動作状態を停止に設定

オーディオCODECのオーディオ・データ用I²Sバス・インタフェース設定

- (a) LRCLKの出力用にタイマ・アレイ・ユニット (TAU) のチャンネル0のTO00出力の設定を行います。
- (b) データ送信 / 受信用にシリアル・アレイ・ユニット0 (SAU0) のチャンネル0のCSI00を設定します。

EEPROMのSPIインタフェース設定

- (a) SPIインタフェースとして使用するCSI受信用にDMAのチャンネル0を設定します。
- (b) SPIインタフェースとして使用するCSI送信用 / 受信時のダミーデータのセット用にDMAのチャンネル1を設定します。
- (c) SPIインタフェースとして使用するシリアル・アレイ・ユニット0 (SAU0) のチャンネル2のCSI10を設定します。
- (d) EEPROMの全消去(オールOFFH)を行います。

キー割り込みを有効にし、キーの取り込みを開始します。

動作中インジケータを消灯します。

割り込みを許可します。

メイン処理へ続きます。

```

;*****
;
;
;      使用する周辺の初期設定
;
;*****
XMAIN   CSEG      UNIT
RESET_START:

;-----
;      割り込み禁止
;-----
DI

;-----
;      レジスタバンク設定
;-----
SEL      RB0

;-----
;      スタック・ポインタの設定
;-----
MOVW    SP,      #LOW  STACKTOP      ;スタック・ポインタを設定

;-----
;      クロック周波数の設定
;-----
;      20MHzのX1発振回路で動作が行えるように設定します
;-----
MOV     CMC,     #01000001B          ;クロック動作モード
;      | | | | | +----- AMPH: 10MHz < fMX 20MHz
;      | | | | | +----- <000>
;      | | | | | +----- OSCSELS: P123/P124端子を入力ポート
;      | | | | | +----- <0>
;      | | | | | +----- EXCLK/OSCSEL: X1発振モード(20MHz)
;      | | | | | +-----

MOV     CSC,     #01000000B          ;クロック動作ステータス制御
;      | | | | | +----- HIOSTOP: 高速内蔵発振回路動作
;      | | | | | +----- <00000>
;      | | | | | +----- XTSTOP: XT1発振回路停止
;      | | | | | +----- MSTOP: X1発振回路動作
;      | | | | | +-----

MOV     OSMC,    #00000001B          ;動作スピード・モード
;      | | | | | +----- FSEL: 10MHzを越える周波数で動作
;      | | | | | +----- <00000>
;      | | | | | +-----

MOV     OSTSS,   #00000101B          ;発振安定時間: 2^15/fX

HRST300:
NOP
BF      OSTC.2,  $HRST300            ;クロック発振安定待ち

MOV     CKC,     #00011000B          ;クロック選択
;      | | | | | +----- MDIV2-0: CPU/周辺ハードウェア・クロック (fCLK) = fMX
;      | | | | | +----- <1>
;      | | | | | +----- MCM0: 高速システム・クロック (fMX)
;      | | | | | +----- <R>
;      | | | | | +----- CSS: メイン・システム・クロック (fMAIN) = fCLK
;      | | | | | +----- <R>
;      | | | | | +-----

;-----
;      ポート0の設定
;-----
MOV     P0,      #00011010B          ;P00, P02, P05-P06の出力ラッチLow
;      | | | | | +----- ;P01, P03-P04の出力ラッチHIGH
;      | | | | | +----- ;P00-P06を出力ポートに設定
;      | | | | | +----- ;P01: オーディオCODECへのLRCLK出力
;      | | | | | +----- ;P02: EEPROMへのSO10出力
;      | | | | | +----- ;P03: EEPROMへのSI10入力
;      | | | | | +----- ;P04: EEPROMへのSCK10出力
;      | | | | | +----- ;P00, P05-P06: 未使用
;      | | | | | +-----

;-----
;      ポート1の設定
;-----
MOV     P1,      #00000000B          ;P10-P17の出力ラッチLow
MOV     PM1,     #00000000B          ;P10-P17を出力ポートに設定
;      | | | | | +----- ;P10: オーディオCODECへのSCK00出力
;      | | | | | +----- ;P11: オーディオCODECへのSI00入力
;      | | | | | +----- ;P12: オーディオCODECへのSO00出力
;      | | | | | +----- ;P13-P17: 未使用
;      | | | | | +-----

;-----
;      ポート2の設定
;-----
MOV     P2,      #00000000B          ;P20-P27の出力ラッチLow
MOV     PM2,     #00000000B          ;P20-P27を出力ポートに設定
;      | | | | | +----- ;P20-P27: 未使用
;      | | | | | +-----

```


;-----			
; ポート3の設定			
;-----			
MOV	P3,	#00000000B	;P30-P31の出力ラッチ _{Low}
MOV	PM3,	#11111100B	;P30-P31を出力ポートに設定
			;P30-P31: 未使用
;-----			
; ポート4の設定			
;-----			
MOV	P4,	#00000000B	;P40-P47の出力ラッチ _{Low}
MOV	PM4,	#00000000B	;P40-P47を出力ポートに設定
			;P40-P47: 未使用
;-----			
; ポート5の設定			
;-----			
MOV	P5,	#00000000B	;P50-P57の出力ラッチ _{Low}
MOV	PM5,	#00000000B	;P50-P57を出力ポートに設定
			;P50-P57: 未使用
;-----			
; ポート6の設定			
;-----			
MOV	P6,	#00000000B	;P60-P67の出力ラッチ _{Low}
MOV	PM6,	#00000000B	;P60-P67を出力ポートに設定
			;P60: オーディオCODEC内のレジスタ設定用のSCL0出力
			;P61: オーディオCODEC内のレジスタ設定用のSDA0入出力
			;P62-P67: 未使用
;-----			
; ポート7の設定			
;-----			
MOV	P7,	#00000000B	;P70-P77の出力ラッチ _{Low}
MOV	PM7,	#00000011B	;P70-P71を入力ポート, P72-P77を出力ポートに設定
			;P70: 再生キー入力
			;P71: 録音キー入力
			;P72: 動作中インジケータ(LED)出力
			;P73: オーディオCODECへのリセット出力
			;P74: EEPROMへのCS0出力
			;P75: EEPROMへのCS1出力
			;P76: EEPROMへのCS2出力
			;P77: EEPROMへのCS3出力
;-----			
; ポート8の設定			
;-----			
MOV	P8,	#00000000B	;P80-P87の出力ラッチ _{Low}
MOV	PM8,	#00000000B	;P80-P87を出力ポートに設定
			;P80-P87: 未使用
;-----			
; ポート11の設定			
;-----			
MOV	P11,	#00000000B	;P110-P111の出力ラッチ _{Low}
MOV	PM11,	#11111100B	;P110-P111を出力ポートに設定
			;P110-P111: 未使用
;-----			
; ポート12の設定			
;-----			
MOV	P12,	#00000000B	;P120の出力ラッチ _{Low}
MOV	PM12,	#11111100B	;P120を出力ポートに設定
			;P120: 未使用
;-----			
; ポート13の設定			
;-----			
MOV	P13,	#00000000B	;P130-P131の出力ラッチ _{Low}
MOV	PM13,	#11111100B	;P131を出力ポートに設定
			;P130-P131: 未使用
;-----			
; ポート14の設定			
;-----			
MOV	P14,	#00000000B	;P140-P145の出力ラッチ _{Low}
MOV	PM14,	#11000000B	;P140-P145を出力ポートに設定
			;P140-P145: 未使用
;-----			
; ポート15の設定			
;-----			
MOV	P15,	#00000000B	;P150-P157の出力ラッチ _{Low}
MOV	PM15,	#00000000B	;P150-P157を出力ポートに設定
			;P150-P157: 未使用
;-----			

```

;-----
; オートオーディオCODEC内のレジスタ設定
;-----
;
; 下記動作を行います。
; ・オーディオCODECへのクロック供給
; ・プログラム内ウエイト用のタイマ設定
; ・オーディオCODECへのリセット信号出力
; ・オーディオCODEC内のレジスタ設定用のI2Cインタフェース設定
;-----
(a)
MOV      RPLAYMOD, #CRESET          ;リセット
; タイマ・アレイユニットのタイマ・クロック選択
SET1     !TAU0EN                    ;タイマ・アレイユニットの入力クロック供給
MOV      !TPS0L, #10010000B        ;動作クロック選択
; |||||++++----- CK00: fCLK
; +----- CK01: fCLK/2^9

;-----
; オートオーディオCODECへのクロック供給
;-----
; TO02出力 (10MHz)
;-----
MOVW     AX, #0000000000000000B;動作モード設定
; |||||++++----- MD023-020: インターバル・タイマ・モード
; |----- <00固定>
; |----- CIS021-020: 未使用
; |----- STS022-020: ソフトウェア・トリガ・スタートのみ有効
; |----- MASTER02: 単体動作
; |----- CSS02: CKS02ビットで指定したマクロ・クロックMCK
; |----- <00固定>
; |----- CKS02: PRSレジスタで設定した動作クロックCK00
; +-----

MOVW     !TMR02, AX
MOVW     TDR02, #0                  ;インターバル設定: 10MHz出力 (20MHz)
SET1     TOE0L.2                    ;TO02出力
SET1     TS0L.2                     ;動作開始 (トリガ・ビット)

;-----
; プログラム内で使用するウエイト用
;-----
; TM01使用
;-----
(b)
MOVW     AX, #1000000000000000B;動作モード設定
; |||||++++----- MD013-010: インターバル・タイマ・モード
; |----- <00固定>
; |----- CIS011-010: 未使用
; |----- STS012-010: ソフトウェア・トリガ・スタートのみ有効
; |----- MASTER01: 単体動作
; |----- CSS01: CKS01ビットで指定したマクロ・クロックMCK
; |----- <00固定>
; |----- CKS01: PRSレジスタで設定した動作クロックCK01
; +-----

MOVW     !TMR01, AX

(c)
;電源安定待ち
MOVW     TDR01, #195*210/5-1        ;インターバル設定 (210ms)
SET1     TS0L.1                     ;タイマ動作開始
CLR1     TMIF01                      ;INTTM00割り込み要求クリア
BF       TMIF01, $$

;リセット
CLR1     P_RESETB
;ウエイト約5us (50ns*5*Bregで算出)
MOV      B, #5000/50/5

HRST550:
DEC      B                           ;1clk
BNZ     $HRST550                     ;4clk (Z=0)
SET1     P_RESETB

;電源安定待ち
MOVW     TDR01, #195*160/5-1        ;インターバル設定 (160ms)
SET1     TS0L.1                     ;タイマ動作開始
CLR1     TMIF01                      ;INTTM00割り込み要求クリア
BF       TMIF01, $$
SET1     TT0L.1                       ;タイマ動作停止

```

```

;-----
; オーディオCODEC内のレジスタ設定値の送信
;-----
; IIC0使用
;-----
(d) SET1    !IIC0EN                ;シリアル・インタフェースIIC0の入カクロック供給
      CLR1    IICE0                ;動作停止
      CLR1    P_SCL                ;SCL0ラッチ
      CLR1    PM_SCL               ;SCL0端子
      CLR1    P_SDA                ;SDA0端子
      CLR1    PM_SDA               ;SDA0端子
      MOV     IICX0, #00000000B    ;転送クロック選択
      ;| | | | | +----- CLX0
      ;+++++++----- <0000000>
      MOV     IICCL0, #00001110B
      ;| | | | | +----- CL01-00(+CLX0): fCLK/96(ファースト・モード)
      ;| | | | | +----- DFC0: デジタル・フィルタ・オン
      ;| | | | | +----- SMC0: ファースト・モードで動作
      ;| | | | | +----- DAD0: <R>SDA0端子のレベル検出
      ;| | | | | +----- CLD0: <R>SCL0端子のレベル検出
      ;| | | | | +----- <00>
      MOV     IICF0, #00000011B   ;通信予約禁止
      ;| | | | | +----- IICRSV: 通信予約機能禁止
      ;| | | | | +----- STCEN: 初期スタート許可
      ;| | | | | +----- <0000固定>
      ;| | | | | +----- IICBSY: <R>IICバス状態フラグ
      ;| | | | | +----- STCF: <R>STT0クリアフラグ
      MOV     IICC0, #00001000B   ;マスタ時の初期設定
      ;| | | | | +----- SPT0: ストップ・コンディション・トリガ
      ;| | | | | +----- STT0: スタート・コンディション・トリガ
      ;| | | | | +----- ACKE0: アクノリッジ制御
      ;| | | | | +----- WTIMO: ウェイトおよび割込要求発生の制御: 9クロック
      ;| | | | | +----- SPIE0: ストップコンディション検出による割込要求発生禁止
      ;| | | | | +----- WRELO: ウェイト解除しない
      ;| | | | | +----- LRELO: 通信退避: 通常動作
      ;| | | | | +----- IICE0: I2Cの動作許可
      SET1    SPT0                ;ストップ・コンディション設定
(e) ; オーディオCODEC内のレジスタ設定: システムON設定
      MOV     ES, #HIGHW TSYSON   ;レジスタ設定値ROMテーブルの先頭アドレス上位4ビット
      MOVW    HL, #LOWW TSYSON    ;下位16ビット
      MOVW    DE, #LOWW TSYSONE  ;レジスタ設定値ROMテーブルの最終アドレス下位8ビット
      CALL    !!SI2CWRITE        ;I2Cライト処理
      MOV     RPLAYMOD, #CSTOP    ;停止

```



```

;-----
; EEPROMの全消去 (オール0FFH)
;-----
;EEPROMライトイネーブル
AND    P_CS,    #00001111B    ;全EEPROMの選択
CLR1   CSIIF10
MOV    SIO10,   #00000110B    ;ライトイネーブル(WREN)命令(INTCSI10発生)
BF     CSIIF10,$$
OR     P_CS,    #11110000B    ;設定完了

;EEPROM 全消去
AND    P_CS,    #00001111B    ;全EEPROMの設定
CLR1   CSIIF10
MOV    SIO10,   #11000111B    ;CE 全消去(INTCSI10発生)
BF     CSIIF10,$$
OR     P_CS,    #11110000B    ;設定完了

SET1   ST0L.2    ;CSI10動作停止(トリガ・ビット)
SET1   CSIMK10  ;INTCSI10割り込み処理禁止

;-----
; キーの取り込み開始
;-----
MOV    KRM,     #00000011B    ;KR0-KR1有効
CLR1   KRIF
SET1   KRMK     ;INTKR割り込み要求クリア
                ;INTKR割り込み処理禁止

SET1   P_LED    ;動作中インジケータOFF(キー入力可能)

;-----
; 割り込み許可
;-----
EI

```

(d)

C言語の使用する周辺の初期設定も、アセンブリ言語と同様な動作を行います。

C言語では、hdwinit 関数を作成することにより、初期設定のタイミングを早くすることができます。

hdwinit 関数は、周辺装置 (sfr) の初期設定をする関数としてユーザが必要に応じて作成する関数です。

```

/*****
使用する周辺の初期設定

*****/
void hdwinit(void)
{
/*-----
割り込み禁止
-----*/
DI();
/*-----
クロック周波数の設定
-----*/
20MHzのX1発振回路で動作が行えるように設定します
-----*/
CMC = 0b01000001; /* クロック動作モード */
/*      ||| ||| +----- AMPH: 10MHz < fMX 20MHz */
/*      ||| ||| +----- <000> */
/*      ||| ||| +----- OSCSELS: P123/P124端子を入力ポート */
/*      ||| ||| +----- <0> */
/*      ++----- EXCLK/OSCSEL: X1発振モード(20MHz) */

CSC = 0b01000000; /* クロック動作ステータス制御 */
/*      ||| ||| +----- HIOSTOP: 高速内蔵発振回路動作 */
/*      ||| ||| +----- <00000> */
/*      ||| ||| +----- XTSTOP: XT1発振回路停止 */
/*      +----- MSTOP: X1発振回路動作 */

OSMC = 0b00000001; /* 動作スピード・モード */
/*      ||| ||| +----- FSEL: 10MHzを越える周波数で動作 */
/*      +----- <00000> */

OSTS = 0b00000101; /* 発振安定時間: 2^15/fX */

while(!OSTC.2) {} /* クロック発振安定待ち */

CKC = 0b00011000; /* クロック選択 */
/*      ||| ||| +----- MDIV2-0: CPU/周辺ハードウェア・クロック (fCLK)=fMX */
/*      ||| ||| +----- <1> */
/*      ||| ||| +----- MCM0: 高速システム・クロック (fMX) */
/*      ||| ||| +----- <R> */
/*      ||| ||| +----- CSS: メイン・システム・クロック (fMAIN)=fCLK */
/*      +----- <R> */

/*-----
ポート0の設定
-----*/
P0 = 0b00011010; /* P00,P02,P05-P06の出力ラッチLow */
/* P01,P03-P04の出力ラッチHIGH */
PM0 = 0b10000000; /* P00-P06を出力ポートに設定 */
/* P01: オーディオCODECへのLRCLK出力 */
/* P02: EEPROMへのSO10出力 */
/* P03: EEPROMへのSI10入力 */
/* P04: EEPROMへのSCK10出力 */
/* P00,P05-P06: 未使用 */

/*-----
ポート1の設定
-----*/
P1 = 0b00000000; /* P10-P17の出力ラッチLow */
PM1 = 0b00000000; /* P10-P17を出力ポートに設定 */
/* P10: オーディオCODECへのSCK00出力 */
/* P11: オーディオCODECへのSI00入力 */
/* P12: オーディオCODECへのSO00出力 */
/* P13-P17: 未使用 */

/*-----
ポート2の設定
-----*/
P2 = 0b00000000; /* P20-P27の出力ラッチLow */
PM2 = 0b00000000; /* P20-P27を出力ポートに設定 */
/* P20-P27: 未使用 */

/*-----
ポート3の設定
-----*/
P3 = 0b00000000; /* P30-P31の出力ラッチLow */
PM3 = 0b11111100; /* P30-P31を出力ポートに設定 */
/* P30-P31: 未使用 */

```

```

/*-----
ポート4の設定
-----*/
P4 = 0b00000000; /* P40-P47の出力ラッチLow */
PM4 = 0b00000000; /* P40-P47を出力ポートに設定 */
/* P40-P47: 未使用 */

/*-----
ポート5の設定
-----*/
P5 = 0b00000000; /* P50-P57の出力ラッチLow */
PM5 = 0b00000000; /* P50-P57を出力ポートに設定 */
/* P50-P57: 未使用 */

/*-----
ポート6の設定
-----*/
P6 = 0b00000000; /* P60-P67の出力ラッチLow */
PM6 = 0b00000000; /* P60-P67を出力ポートに設定 */
/* P60: オーディオCODEC内のレジスタ設定のためのSCL0出力 */
/* P61: オーディオCODEC内のレジスタ設定のためのSDA0入出力 */
/* P62-P67: 未使用 */

/*-----
ポート7の設定
-----*/
P7 = 0b00000000; /* P70-P77の出力ラッチLow */
PM7 = 0b00000011; /* P70-P71を入力ポート, P72-P77を出力ポートに設定 */
/* P70: 再生キー入力 */
/* P71: 録音キー入力 */
/* P72: 動作中インジケータ (LED) 出力 */
/* P73: オーディオCODECへのリセット出力 */
/* P74: EEPROMへのCS0出力 */
/* P75: EEPROMへのCS1出力 */
/* P76: EEPROMへのCS2出力 */
/* P77: EEPROMへのCS3出力 */

/*-----
ポート8の設定
-----*/
P8 = 0b00000000; /* P80-P87の出力ラッチLow */
PM8 = 0b00000000; /* P80-P87を出力ポートに設定 */
/* P80-P87: 未使用 */

/*-----
ポート11の設定
-----*/
P11 = 0b00000000; /* P110-P111の出力ラッチLow */
PM11 = 0b11111100; /* P110-P111を出力ポートに設定 */
/* P110-P111: 未使用 */

/*-----
ポート12の設定
-----*/
P12 = 0b00000000; /* P120の出力ラッチLow */
PM12 = 0b11111110; /* P120を出力ポートに設定 */
/* P120: 未使用 */

/*-----
ポート13の設定
-----*/
P13 = 0b00000000; /* P130-P131の出力ラッチLow */
PM13 = 0b11111100; /* P131-P131を出力ポートに設定 */
/* P130-P131: 未使用 */

/*-----
ポート14の設定
-----*/
P14 = 0b00000000; /* P140-P145の出力ラッチLow */
PM14 = 0b11000000; /* P140-P145を出力ポートに設定 */
/* P140-P145: 未使用 */

/*-----
ポート15の設定
-----*/
P15 = 0b00000000; /* P150-P157の出力ラッチLow */
PM15 = 0b00000000; /* P150-P157を出力ポートに設定 */
/* P150-P157: 未使用 */

```



```

/*-----
オーディオCODEC内のレジスタ設定
-----*/

下記動作を行います。
・オーディオCODECへのクロック供給
・プログラム内ウエイト用のタイマ設定
・オーディオCODECへのリセット信号出力
・オーディオCODEC内のレジスタ設定用のI2Cインタフェース設定
-----*/

/* タイマ・アレイユニットのタイマ・クロック選択 */
TAU0EN = 1; /* タイマ・アレイユニットの入力クロック供給 */
TPS0L = 0b10010000; /* 動作クロック選択 */
/*      ||| |++++----- CK00: fCLK */
/*      +----- CK01: fCLK/2^9 */

/*-----
オーディオCODECへのクロック供給
-----*/

TO02出力 (10MHz)
-----*/
TMR02 = 0b0000000000000000; /* 動作モード設定 */
/*      ||| |++++----- MD023-020: インターバル・タイマ・モード */
/*      +----- <00固定> */
/*      ||| |++++----- CIS021-020: 未使用 */
/*      +----- STS022-020: ソフトウェア・トリガ・スタートのみ有効 */
/*      +----- MASTER02: 単体動作 */
/*      +----- CSS02: CKS02ビットで指定したマクロ・クロックMCK */
/*      +----- <00固定> */
/*      +----- CKS02: PRSレジスタで設定した動作クロックCK00 */

TDR02 = 0; /* インターバル設定: 10MHz出力 (20MHz) */
TOE0L.2 = 1; /* TO02出力 */
TS0L.2 = 1; /* 動作開始 (トリガ・ビット) */

/*-----
プログラム内で使用するウエイト用
-----*/

TM01使用
-----*/
TMR01 = 0b1000000000000000; /* 動作モード設定 */
/*      ||| |++++----- MD013-010: インターバル・タイマ・モード */
/*      +----- <00固定> */
/*      ||| |++++----- CIS011-010: 未使用 */
/*      +----- STS012-010: ソフトウェア・トリガ・スタートのみ有効 */
/*      +----- MASTER01: 単体動作 */
/*      +----- CSS01: CKS01ビットで指定したマクロ・クロックMCK */
/*      +----- <00固定> */
/*      +----- CKS01: PRSレジスタで設定した動作クロックCK01 */

/* 電源安定待ち */
TDR01 = 195*210/5-1; /* インターバル設定 (210ms) */
TS0L.1 = 1; /* タイマ動作開始 */
TMIF01 = 0; /* INTTM00割り込み要求クリア */
while(!TMIF01) {}

/* リセット */
P_RESETB = 0;
/* ウエイト約5us以上 */
TDR01 = 0; /* インターバル設定 */
TS0L.1 = 1; /* タイマ動作開始 */
TMIF01 = 0; /* INTTM00割り込み要求クリア */
while(!TMIF01) {}
P_RESETB = 1;

/* 電源安定待ち */
TDR01 = 195*160/5-1; /* インターバル設定 (160ms) */
TS0L.1 = 1; /* タイマ動作開始 */
TMIF01 = 0; /* INTTM00割り込み要求クリア */
while(TMIF01) {}
TT0L.1 = 1; /* タイマ動作停止 */

/*-----
オーディオCODEC内のレジスタ設定値の送信
-----*/

IIC0使用
-----*/
IIC0EN = 1; /* シリアル・インタフェースIIC0の入力クロック供給 */

IICE0 = 0; /* 動作停止 */

P_SCL = 0; /* SCL0ラッチ */
PM_SCL = 0; /* SCL0端子 */
P_SDA = 0; /* SDA0端子 */
PM_SDA = 0; /* SDA0端子

```

```

IICX0 = 0b00000000; /* 転送クロック選択 */
/*      |||||+----- CLX0 */
/*      +----- <0000000> */
IICCL0 = 0b00001110;
/*      |||||+----- CL01-00(+CLX0): fCLK/96(ファースト・モード) */
/*      |||||+----- DFC0: デジタル・フィルタ・オン */
/*      |||||+----- SMC0: ファースト・モードで動作 */
/*      |||||+----- DAD0: <R>SDA0端子のレベル検出 */
/*      |||||+----- CLD0: <R>SCL0端子のレベル検出 */
/*      +----- <00> */

IICF0 = 0b00000011; /* 通信予約禁止 */
/*      |||||+----- IICRSV: 通信予約機能禁止 */
/*      |||||+----- STCEN: 初期スタート許可 */
/*      |||||+----- <0000固定> */
/*      |||||+----- IICBSY: <R>IICバス状態フラグ */
/*      |||||+----- STCF: <R>STT0クリアフラグ */

IICC0 = 0b00001000; /* マスタ時の初期設定 */
/*      |||||+----- SPT0: ストップ・コンディション・トリガ */
/*      |||||+----- STT0: スタート・コンディション・トリガ */
/*      |||||+----- ACKE0: アクノリッジ制御 */
/*      |||||+----- WTIMO: ウェイトおよび割込要求発生の制御:9クロック */
/*      |||||+----- SPIE0: ストップコンディション検出による割込要求発生禁止 */
/*      |||||+----- WRELO: ウェイト解除しない */
/*      |||||+----- LRELO: 通信退避:通常動作 */
/*      |||||+----- IICE0: I2Cの動作許可 */

SPT0 = 1; /* ストップ・コンディション設定 */

/* オーディオCODEC内のレジスタ設定: システムON設定 */
fn_I2cWrite(&aSystemOnTbl[0][0], sizeof(aSystemOnTbl));

/*-----*/
オーディオCODECのオーディオ・データ用I2Sバス・インタフェース設定
/*-----*/

下記動作を行います。
・LRCLK出力用にTO00出力とINTFM00割り込みの設定
・データ送受信用にCSI00の設定
/*-----*/

/*-----*/
LRCLKの出力設定
/*-----*/
TO00出力(16kHz)
/*-----*/
TMR00 = 0b0000000000000000; /* 動作モード設定 */
/*      |||||+----- MD003-000: インターバル・タイマ・モード */
/*      |||||+----- <00固定> */
/*      |||||+----- CIS001-000: 未使用 */
/*      |||||+----- STS002-000: ソフトウェア・トリガ・スタートのみ有効 */
/*      |||||+----- MASTER00: 単体動作 */
/*      |||||+----- CSS00: CKS00ビットで指定したマクロ・クロックMCK */
/*      |||||+----- <00固定> */
/*      |||||+----- CKS00: PRSレジスタで設定した動作クロックCK00 */
TDR00 = 625-1; /* インターバル設定: 16kHz出力(32KHz) */

TMPR100 = 0; /* 優先順位を最高位に設定 */
TMPR000 = 0;

/*-----*/
CSI00の設定
/*-----*/
SAU0EN = 1; /* シリアル・アレイ・ユニットの入力クロック供給 */

NOP(); /* ウェイト */
NOP();
NOP();
NOP();

SPS0L = 0b00000000; /* 動作クロック選択 fCLK */
/*      |||||+----- PRS003-00: fCLK */
/*      |||||+----- PRS013-10: 未使用 */

SDR00 = (12-1) << 9; /* ビット15-7: 転送クロック設定(833kHz) */

SMR00 = 0b0000000000100001; /* 動作モード選択: CSIモード */
/*      |||||+----- MD000: バッファ空き割り込み */
/*      |||||+----- MD002-1: CSIモード */
/*      |||||+----- <100固定> */
/*      |||||+----- SIS000: 未使用 */
/*      |||||+----- <0固定> */
/*      |||||+----- STS00: ソフトウェア・トリガのみ有効(CSI時は固定) */
/*      |||||+----- <00000固定> */
/*      |||||+----- CSS00: 転送クロックをCKS00ビットで指定した動作クロック */
/*      |||||+----- MCKの分周クロックに設定 */
/*      |||||+----- CKS00: 動作クロックをPRSレジスタで設定したプリスケラ */
/*      |||||+----- 出力クロックCK00に設定 */

```

```

/* 初期データ出力 (CSI10の設定時に同時に行う) */
/* SO0 = 0b0000000100000001; /* SO端子&SCK端子の初期出力設定 */
/*      |||||+++++----- SO0n-0: 送信で使用 */
/*      +++++----- CKOn-0: チャンネルn-0のシリアル・クロック出力 */

SOE0L.0 = 1;          /* 出力許可 */

P_SCK00 = 1;          /* SCK00ラッチ:ハイ */
PM_SCK00 = 0;         /* SCK00端子出力設定 */
PM_SI00 = 1;          /* SI00端子入力設定 */
P_SO00 = 1;           /* SO00ラッチ:ハイ */
PM_SO00 = 0;         /* SO00端子出力設定 */

SS0L.0 = 1;          /* CSI00動作開始(トリガ・ビット) */
CSIIF00 = 0;         /* INTCSI00割り込み要求クリア */
CSIMK00 = 0;         /* INTCSI00割り込み処理許可 */

CSIPR100 = 0;        /* 優先順位をINTTM00の次に設定 */
CSIPR000 = 1;

/*-----
EEPROM用のSPIインタフェース設定
-----*/

下記動作を行います。
・データ送信用にCSI10の設定
・連続送信用として受信用にDMA0, 送信用にDMA1を設定

-----*/
P_CS |= 0b011110000; /* 全EEPROMをノン・アクティブ */

/*-----
DMA0の設定 (CSI受信用)
-----*/
DEN0 = 1;            /* DMAチャンネル0の動作許可 */

DSA0 = 0x044;        /* DMA_SFRアドレス: SDR02(SIO10)=0FFF44H */
DMC0 = 0b00001000; /* DMAチャンネル0の転送モード設定 */
/*      ||++++----- IFC03-0: DMA起動要因:INTCSI10転送完了割り込み */
/*      +----- DWAIT0: DMA転送の保留:DMA起動要求によりDMA転送を行う
/*      (保留しない) */
/*      +----- DS0: 転送データ・サイズ:8ビット */
/*      +----- DRS0: DMA転送方向の選択:SFR 内蔵RAM */
/*      +----- STG0: ソフトウェア・トリガ動作しない */

/*-----
DMA1の設定
(CSI送信用, CSI受信時のダミー・データのセット用)
-----*/
DEN1 = 1;            /* DMAチャンネル1の動作許可 */

DSA1 = 0x044;        /* DMA_SFRアドレス: SDR02(SIO10)=0FFF44H */
DMC1 = 0b01001000; /* DMAチャンネル0の転送モード設定 */
/*      ||++++----- IFC13-0: DMA起動要因:INTCSI10転送完了割り込み */
/*      +----- DWAIT1: DMA転送の保留:DMA起動要求によりDMA転送を行う
/*      (保留しない) */
/*      +----- DS1: 転送データ・サイズ:8ビット */
/*      +----- DRS1: DMA転送方向の選択:内蔵RAM SFR */
/*      +----- STG1: ソフトウェア・トリガ動作しない */

/*-----
CSI10の設定
-----*/
SDR02 = (4-1) << 9; /* ビット15-7: 転送クロック設定(2.5MHz) */

SMR02 = 0b000000000100000; /* 動作モード選択:CSIモード */
/*      ||+----- MD020: 転送完了割り込み */
/*      ||+----- MD022-1: CSIモード */
/*      ||+----- <100固定> */
/*      ||+----- SIS020: 未使用 */
/*      ||+----- <0固定> */
/*      ||+----- STS02: ソフトウェア・トリガのみ有効(CSI時は固定) */
/*      ||+----- <00000固定> */
/*      ||+----- CSS02: 転送クロックをCKS00ビットで指定した動作クロック
/*      MCKの分周クロックに設定 */
/*      ||+----- CKS02: 動作クロックをPRSレジスタで設定したプリスケアラ
/*      出力クロックCK00に設定 */
SCR02 = 0b1011000000000111; /* 通信フォーマットの設定 */
/*      ||+----- DLS022-0: 8ビット・データ長 */
/*      ||+----- <0固定> */
/*      ||+----- SLC021-0: 未使用(0固定) */
/*      ||+----- <0固定> */
/*      ||+----- DIR02: MSBファーストで入出力を行う */
/*      ||+----- PTC021-0: 未使用(00固定) */
/*      ||+----- EOC02: 未使用(0固定) */
/*      ||+----- <0固定> */
/*      ||+----- CKP02/DAP02: CSIモードでのデータとクロックの位相選択 */
/*      [11選択] */
/*      ||+----- TXE00/RXE00: 送信のみを行う */

```

```

/* 初期データ出力 */
SO0 = 0b0000000100000001; /* SO端子&SCK端子の初期出力設定 */
/*      |||||+++++----- SO0n-0: 送信で使用 */
/*      +++++----- CKOn-0: チャネルn-0のシリアル・クロック出力 */
SOEOL.2 = 1; /* 出力許可 */

P_SCK10 = 1; /* SCK00ラッチ:ハイ */
PM_SCK10 = 0; /* SCK00端子出力設定 */
PM_SII10 = 1; /* SII00端子入力設定 */
P_SO10 = 1; /* SO00ラッチ:ハイ */
PM_SO10 = 0; /* SO00端子出力設定 */

SSOL.2 = 1; /* CSI10動作開始(トリガ・ビット) */

/*-----
EEPROMの全消去(オールOFFH)
-----*/
/* EEPROMライトイネーブル */
P_CS &= 0b00001111; /* 全EEPROMの選択 */
CSIIF10 = 0;
SIO10 = 0b00000110; /* 送信データ・セット(ライトイネーブル(WREN)命令, INTCSI10発生) */
while(!CSIIF10) {}
P_CS |= 0b11110000; /* 設定完了 */

/* EEPROM 全消去 */
P_CS &= 0b00001111; /* 全EEPROMの選択 */
CSIIF10 = 0;
SIO10 = 0b11000111; /* 送信データ・セット(CE 全消去, INTCSI10発生) */
while(!CSIIF10) {}
P_CS |= 0b11110000; /* 設定完了 */

ST0L.2 = 1; /* CSI10動作停止(トリガ・ビット) */
CSIMK10 = 1; /* INTCSI10割り込み処理禁止 */

/*-----
キーの取り込み開始
-----*/
KRM = 0b00000011; /* KR0-KR1有効 */
KRIF = 0; /* INTKR割り込み要求クリア */
KRMK = 1; /* INTKR割り込み処理禁止 */

P_LED = 1; /* 動作中インジケータOFF(キー入力可能) */

/*-----
割り込み許可
-----*/
EI();
}

```

4.4 メイン処理

アセンブリ言語のメイン処理では、次の動作を行います。

キー処理を行います。

のキー処理の判定で録音キーが押下された場合、キー入力を無効にし、録音動作を開始します。

のキー処理の判定で再生キーが押下された場合、キー入力を無効にし、再生動作を開始します。

動作状態が録音終了の場合、録音動作を終了させます。

動作状態が再生終了の場合、再生動作を終了させます。

動作状態を停止とし、キー入力を有効にします。

の処理へ


```

; -----
; 録音終了設定
; -----
LMAINRECE:
CMP      RPLAYMOD, #CREC_END          ; 録音終了中?
BNZ      $LMAINPLAYE                 ; No,

; オーディオCODEC内のレジスタ設定: 録音終了設定
MOV      ES,      #HIGHW TRECOFF     ; レジスタ設定値ROMテーブルの先頭アドレス上位4ビット
MOVW    HL,      #LOWW TRECOFF      ; 下位16ビット
MOVW    DE,      #LOWW TRECOFFE     ; レジスタ設定値ROMテーブルの最終アドレス下位8ビット
CALL    !!SI2CWRITE                  ; I2Cライト処理
BR      LMAINSTOP

; -----
; 再生終了設定
; -----
LMAINPLAYE:
CMP      RPLAYMOD, #CPLAY_END        ; 再生終了中?
BNZ      $LMAINRET                   ; No,

; オーディオCODEC内のレジスタ設定: 再生終了設定
MOV      ES,      #HIGHW TPLAYOFF    ; レジスタ設定値ROMテーブルの先頭アドレス上位4ビット
MOVW    HL,      #LOWW TPLAYOFF     ; 下位16ビット
MOVW    DE,      #LOWW TPLAYOFFE    ; レジスタ設定値ROMテーブルの最終アドレス下位8ビット
CALL    !!SI2CWRITE                  ; I2Cライト処理

LMAINSTOP:
MOV      KRM,     #00000011B         ; KR0-KR1有効
MOV      RPLAYMOD, #CSTOP            ; 停止
SET1    P_LED                    ; 動作中インジケータOFF (キー入力可能)

LMAINRET:
BR      MAIN_LOOP

```


4.6 INTCSI00割り込み処理

アセンブリ言語のINTCSI00割り込み処理では、次の動作を行います。

レジスタ・バンクを切り替えます。

受信したデータの保存先アドレスが1ページに達したかのチェックを行い、1ページに達した場合、録音データのEEPROMへのライト動作を開始します。

録音中の1バイト毎（1バイト目と2バイト目）のCSI00受信動作です。

再生中の1バイト毎（2バイト目と3バイト目）のCSI00送信動作です。

送信したデータの参照先アドレスが1ページに達したかのチェックを行い、1ページに達した場合、EEPROMの選択を切り替え、EEPROMからのリード動作を開始します。


```

HI2SR500:
MOVW    HL,      RI2SADDR      ;I2S受信データの保存先アドレス
MOV     [HL],    A              ;受信データ保存
HI2SR800:
INCW    RI2SADDR      ;I2S受信データの保存先アドレスの更新
BR      HI2SRET

; -----
; 再生中
; -----
HI2SPLAY:
CMP     RPLAYMOD, #CPLAY      ;再生中?
BNZ     $HI2SRET              ; No,

CMP     RCSI00CNT, #3         ;LRCLKとの同期タイミング?
BNC     $HI2SP100             ; (3バイト目の送信動作開始直後?)

MOVW    HL,      RI2SADDR      ;I2S送信データの保存先アドレス
MOV     A,       [HL]          ;送信データ(2バイト目もしくは3バイト目)
MOV     SIO00,   A            ;送信データ・セット(INTCSI00発生)
INCW    RI2SADDR      ;I2S送信データの保存先アドレスの更新
BR      HI2SP200

HI2SP100:
MOV     RCSI00CNT, #0         ;CSI00送信カウンタ初期化
HI2SP200:
MOVW    AX,      RI2SADDR      ;I2S送信データの保存先アドレス
BT      FI2SPAGE, $HI2SP300    ;ページ2使用中? Yes,
CMPW    AX,      #RRECMEM1E    ;ページ1の最終?
BC      $HI2SRET              ; No,
MOVW    RI2SADDR, #RRECMEM2    ;I2S送信データの保存先アドレス: ページ2
SET1    FI2SPAGE
BR      HI2SP500

HI2SP300:
CMPW    AX,      #RRECMEM2E    ;ページ2の最終?
BC      $HI2SRET              ; No,
MOVW    RI2SADDR, #RRECMEM1    ;I2S送信データの保存先アドレス: ページ1
CLR1    FI2SPAGE

HI2SP500:
;EEPROMへのリード開始
MOV     A,       P_CS          ;EEPROM選択
AND     A,       #00001111B
OR      A,       RP_CS
MOV     P_CS,   A
MOV     REEPSEQ, #CEEPSEQ_INST ;命令バイト送信
CLR1    CSIF10                ;INTCSI10割り込み要求クリア
CLR1    CSIMK10               ;INTCSI10割り込み処理許可
SET1    SS0L.2                ;CSI10動作開始(トリガ・ビット)
MOV     SIO10,   #00000011B    ;送信データ(リード指定)・セット(INTCSI10発生)

HI2SRET:
RETI
    
```



```

if(ucI2sByteCouter==1){
/* 動作開始直後のバッファ空き割り込み */
SIO00 = 0x0FF; /* ダミー・データのセット */
}
else{
if(ucI2sByteCouter==2){
/* 1バイト目の受信 */
*ucI2sAddress = SIO00; /* I2S受信データの受信と保存 */
SIO00 = 0x0FF; /* ダミー・データのセット */
}
else/*if(ucI2sByteCouter==3)*/{
/* 2バイト目の受信 */
*ucI2sAddress = SIO00; /* I2S受信データの受信と保存(データの受信のみ行う) */
ucI2sByteCouter = 0; /* CSI00受信カウンタ初期化 */
}
ucI2sAddress++;
}
}
else if(ucPlayMode==CPLAY){
/*-----
再生中
-----*/
if(ucI2sByteCouter<3){
/* 2バイト目もしくは3バイト目の送信 */
SIO00 = *ucI2sAddress; /* 送信データ・セット(I2S送信データ, INTCSI00発生) */
ucI2sAddress++; /* I2S送信データの保存先アドレスの更新 */
}
else/*if(ucI2sByteCouter==3)*/{
/* LRCLKとの同期タイミング(3バイト目の送信動作開始直後) */
ucI2sByteCouter = 0; /* CSI00送信カウンタ初期化 */
}

if(bI2sMemoryPage){
/* ページ2使用中 */
if(ucI2sAddress>=&ucMemoryPage2[3*2*42]){
/* ページ2の最終 */
ucI2sAddress = &ucMemoryPage1[0];
/* I2S送信データの保存先アドレス: ページ1 */
bI2sMemoryPage = 0;
/* EEPROMへのリード開始 */
P_CS = (P_CS & 0b00001111)|ucEepromCs; /* EEPROM選択 */
ucEepromSeq = CEEPSEQ_INST; /* 命令バイト送信 */
CSIIF10 = 0; /* INTCSI10割り込み要求クリア */
CSIMK10 = 0; /* INTCSI10割り込み処理許可 */
SS0L.2 = 1; /* CSI10動作開始(トリガ・ビット) */
SIO10 = 0b00000011; /* 送信データ・セット(リード指定, INTCSI10発生) */
}
}
else{
/* ページ1使用中 */
if(ucI2sAddress>=&ucMemoryPage1[3*2*42]){
/* ページ1の最終 */
ucI2sAddress = &ucMemoryPage2[0];
/* I2S送信データの保存先アドレス: ページ2 */
bI2sMemoryPage = 1;
/* EEPROMへのリード開始 */
P_CS = (P_CS & 0b00001111)|ucEepromCs; /* EEPROM選択 */
ucEepromSeq = CEEPSEQ_INST; /* 命令バイト送信 */
CSIIF10 = 0; /* INTCSI10割り込み要求クリア */
CSIMK10 = 0; /* INTCSI10割り込み処理許可 */
SS0L.2 = 1; /* CSI10動作開始(トリガ・ビット) */
SIO10 = 0b00000011; /* 送信データ・セット(リード指定, INTCSI10発生) */
}
}
}
}
}
}

```

4.7 INTCSI10割り込み処理

アセンブリ言語のINTCSI10割り込み処理では、次の動作を行います。

多重割り込みを許可し、レジスタ・バンクを切り替えます。

録音中のEEPROMへのデータ・ライト動作の命令バイトの送信部分です。

録音中のEEPROMへのデータ・ライト動作の24ビット・アドレスの送信部分です。1バイトずつの3回に分け送信します。

録音中のEEPROMへのデータ・ライト動作のデータ部分の送信開始タイミングとなります。DMAを使用した送信動作を開始します。

再生中のEEPROMからのデータ・リード動作の24ビット・アドレスの送信部分です。1バイトずつの3回に分け送信します。

再生中のEEPROMへのデータ・リード動作のデータ部分の受信開始タイミングとなります。DMAを使用した受信動作を開始します。

```

;*****
;
;
;   INTCSI10割り込み処理
;   (INTCSI10使用, CSI送信 / 受信用)
;
;-----
;
;
;   録音中は, EEPROMへの命令バイトの送信と24ビット・アドレスの送信を行い,
;   EEPROMへのデータ送信のタイミングで, DMAのチャンネル1を開始し, CSI10の
;   シングル送信にて続けて送信します。
;
;   再生中は, EEPROMへの24ビット・アドレスの送信を行い, EEPROMからのデー
;   タ受信のタイミングで, DMAのチャンネル0とチャンネル1を開始し, CSI10のシ
;   ングル受信にて続けて送信します。ただし, DMAのチャンネル1はダミー・デー
;   タのセット用となります。
;
;   レジスタバンク1使用
;
;*****
IINTCSI10:
EI                                     ;多重割り込みの許可
SEL      RB2
;
INC      REEPSEQ                       ;EEPROMの転送シーケンス更新
;
CMP      RPLAYMOD, #CREC               ;録音中? No, (再生中)
BZ      $HCSI1REC
BR      HCSI1PLAY
;
HCSI1REC:
;-----
;   録音中
;-----
;   命令バイト送信
CMP      REEPSEQ, #CEEPSEQ_INST       ;命令バイト送信タイミング?
BNZ     $HCSI1100                     ; No,
OR      P_CS, #11110000B              ; ノン・アクティブ
NOP
MOV     A, P_CS                       ;EEPROM選択
AND    A, #00001111B
OR     A, RP_CS
MOV    P_CS, A
MOV    A, #00000010B                   ;ライト指定
BR     HCSI1650                         ;データ送信
;
HCSI1100:
;   24ビット・アドレス上位送信
CMP      REEPSEQ, #CEEPSEQ_ADDRH     ;24ビット・アドレス上位送信タイミング?
BNZ     $HCSI1200                     ; No,
MOV     A, REEPADDR+1
BR     HCSI1650                         ;データ送信
;
HCSI1200:
;   24ビット・アドレス中位送信
CMP      REEPSEQ, #CEEPSEQ_ADDRM     ;24ビット・アドレス中位送信タイミング?
BNZ     $HCSI1300                     ; No,
MOV     A, REEPADDR
BR     HCSI1650                         ;データ送信
;
HCSI1300:
;   24ビット・アドレス下位送信
CMP      REEPSEQ, #CEEPSEQ_ADDRLL    ;24ビット・アドレス下位送信タイミング?
BNZ     $HCSI1400                     ; No,
MOV     A, #000H
BR     HCSI1650                         ;データ送信
;
HCSI1400:
;   データ送信開始
;   DMA転送開始
BT      FI2SPAGE, $HCSI1430           ;I2Sはページ2を使用中?
MOVW   DRA1, #RRECMEM2+1             ;送信バッファ
MOV    A, !RRECMEM2                  ;第一送信データ
BR     HCSI1450
;
HCSI1430:
MOVW   DRA1, #RRECMEM1+1             ;送信バッファ
MOV    A, !RRECMEM1                  ;第一送信データ
;
HCSI1450:
MOVW   DBC1, #3*2*42-1                ;1ページ分
SET1   DST1                           ;DMA1転送許可 (CSI10送信用)
;
CLR1   DMAIF1                          ;INTDM1割り込み要求クリア
CLR1   DMAMK1                          ;INTDM1割り込み処理許可
CLR1   CSIIF10                          ;INTCSI10割り込み要求クリア
SET1   CSIMK10                          ;INTCSI10割り込み処理禁止
;
SET1   DMAMK0                           ;INTDM0割り込み処理禁止
;
HCSI1650:
SET1   SS0L.2                           ;CSI00動作開始 (トリガ・ビット)
MOV    SIO10, A                        ;送信データ・セット (INTCSI00発生)
;
HCSI1800:
BR     HCSI1RET

```


C言語の処理も、アセンブリ言語と同様な動作を行います。

```

/*****
INTCSI10割り込み処理
(INTCSI10使用, CSI送信 / 受信用)

-----

録音中は, EEPROMへの命令バイトの送信と24ビット・アドレスの送信を行い,
EEPROMへのデータ送信のタイミングで, DMAのチャンネル1を開始し, CSI10の
シングル送信にて続けて送信します。
再生中は, EEPROMへの24ビット・アドレスの送信を行い, EEPROMからのデー
タ受信のタイミングで, DMAのチャンネル0とチャンネル1を開始し, CSI10のシン
グル受信にて続けて受信します。ただし, DMAのチャンネル1はダミー・データ
のセット用となります。

レジスタバンク1使用

*****/
__interrupt void fn_intcsi10(void)
{
    EI(); /* 多重割り込みを許可 */

    if(ucPlayMode==CREC){
        /*-----
        録音中
        -----*/
        ucEepromSeq++; /* EEPROMの転送シーケンス更新 */
        /* 命令バイト送信 */
        switch(ucEepromSeq){
            case CEEPSEQ_INST:
                /* 命令バイト送信タイミング */
                P_CS |= 0b11110000; /* ノン・アクティブ */
                NOP();
                P_CS = (P_CS & 0b00001111)|ucEepromCs; /* EEPROM選択 */
                SIO10 = 0b00000010; /* ライト指定送信 */
                break;

            case CEEPSEQ_ADDRH:
                /* 24ビット・アドレス上位送信タイミング */
                SIO10 = (unsigned char)(ushEepromAddress>>8);
                break;

            case CEEPSEQ_ADDRM:
                /* 24ビット・アドレス中位送信タイミング */
                SIO10 = (unsigned char)(ushEepromAddress);
                break;

            case CEEPSEQ_ADDRL:
                /* 24ビット・アドレス下位送信タイミング */
                SIO10 = 0x000;
                break;

            case CEEPSEQ_DATA:
            default:
                /* データ送信開始 */
                /* DMA転送開始 */
                if(bI2sMemoryPage){
                    /* ページ2使用中 */
                    DRA1 = (unsigned short)&ucMemoryPage1[1];
                    /* 送信バッファをページ1に設定 */
                }
                else{
                    /* ページ1使用中 */
                    DRA1 = (unsigned short)&ucMemoryPage2[1];
                    /* 送信バッファをページ2に設定 */
                }
                DBC1 = 3*2*42-1; /* 1ページ分 */
                DST1 = 1; /* DMA1転送許可(CSI10送信用) */

                DMAIF1 = 0; /* INTDM1割り込み要求クリア */
                DMAMK1 = 0; /* INTDM1割り込み処理許可 */
                CSIIF10 = 0; /* INTCSI10割り込み要求クリア */
                SIMK10 = 1; /* INTCSI10割り込み処理禁止 */
                DMAMK0 = 1; /* INTDM0割り込み処理禁止 */
                SS0L.2 = 1; /* CSI00動作開始(トリガ・ビット) */

                if(bI2sMemoryPage){
                    /* ページ2使用中 */
                    SIO10 = *ucMemoryPage1; /* 送信データ・セット(INTCSI00発生) */
                }
                else{
                    SIO10 = *ucMemoryPage2; /* 送信データ・セット(INTCSI00発生) */
                    /* ページ1使用中 */
                }
                break;
        }
    }
}

```


4.8 INTDMA0割り込み処理

アセンブリ言語のINTDMA0割り込み処理では、次の動作を行います。

レジスタ・バンクを切り替えます。

CSI10の通信動作が終了するまでウェイトします。

動作状態が再生準備開始の場合、動作状態を再生準備とし、2ページ目のEEPROMからのリード動作を開始します。

動作状態が再生準備の場合、2ページ分の再生用データが取得できたこととなりますので、動作状態を再生中とし、I²Sバス・インタフェースの動作を開始し、オーディオCODECへデータの送信を行います。

動作状態が再生の場合、EEPROMの選択を行います。EEPROM全ページのデータのリード動作が終了した場合、I²Sバス・インタフェースの動作を停止し、動作状態を再生停止とします。

DMAの動作を停止します。

```

;*****
;
;   INTDMA0割り込み処理
;   (INTDMA0使用, CSI受信用)
;
;-----
;
;   再生中, EEPROMから1ページ分のデータを受信完了したときに発生する割り込み
;   処理です。使用する4個のEEPROMの選択, I2Sバス・インタフェースの動作開始と
;   動作停止を行います。
;
;   レジスタバンク1使用
;
;*****
IINTDMA0:
-----
SEL      RB1
;通信動作の終了待ち
HDMA0100:
MOV      A,          SSR02L          ;通信動作完了?
BT       A.6,        $HDMA0100      ; No,

;1ページ分のデータ受信完了
OR       P_CS,       #11110000B     ;ノン・アクティブ
MOV      REEPSEQ, #CEEPSEQ_INST

CMP      RPLAYMOD, #CPLAY            ;再生中?
BZ      $HDMA0500                    ; Yes,
CMP      RPLAYMOD, #CPLAY_SET        ;再生準備?
BZ      $HDMA0300                    ; Yes,
CMP      RPLAYMOD, #CPLAY_START      ;再生準備開始?
BNZ     $HDMA0RET                    ; No,

;EEPROMからのリード開始
MOV      RPLAYMOD, #CPLAY_SET        ;再生準備
MOV      A,          P_CS            ;EEPROM選択
AND      A,          #00001111B
OR       A,          #11010000B
MOV      P_CS,      A
CLR1    FI2SPAGE
CLR1    CSIIIF10                    ;INTCSI10割り込み要求クリア
CLR1    CSIMK10                     ;INTCSI10割り込み処理許可
SET1    SS0L.2                      ;CSI10動作開始(トリガ・ビット)
MOV      SIO10,     #00000011B      ;送信データ(リード指定)・セット(INTCSI10発生)
BR      HDMA0RET

;-----
; I2Sバス・インタフェース動作開始
;-----
HDMA0300:
MOV      RPLAYMOD, #CPLAY            ;再生中
MOVW    RI2SADDR, #RRECMEM1+1       ;送信バッファの先頭を設定
CLR1    FI2SPAGE
MOV      A,          !RRECMEM1
;I2Sバス・インタフェース動作開始
SET1    SS0L.0                      ;CSI00動作開始(トリガ・ビット)
CLR1    P_LRCLK                      ;LRCLKの動作開始
SET1    TS0L.0                      ;TO00出力動作開始(トリガ・ビット)
MOV      SIO00,     A                ;送信データ・セット(INTCSI00発生)

BR      HDMA0RET

HDMA0500:
;EEPROMの選択
MOV      A,          RP_CS
SET1    A.3
ROL     A,          1
BC      $HDMA0700                    ;EEPROMは一周した? No,
INCW    REEPADDR
MOVW    AX,          REEPADDR
CMPW    AX,          #128000/256     ;EEPROMの終端?
BNC     $HDMA0800                    ; Yes,
MOV      A,          #11100000B

HDMA0700:
AND     A,          #11110000B
MOV     RP_CS,     A
BR      HDMA0RET

HDMA0800:
;I2Sバス・インタフェース動作停止
SET1    P_LRCLK                      ;LRCLK 出力無効化(ハイ)
SET1    TT0L.0                      ;TO00出力動作停止(トリガ・ビット)
SET1    ST0L.0                      ;CSI00出力動作停止(トリガ・ビット)
SET1    ST0L.2                      ;CSI010出力動作停止(トリガ・ビット)
MOV     RPLAYMOD, #CPLAY_END        ;再生終了中

HDMA0RET:
CLR1    DST1                      ;DMA1転送停止
CLR1    DST0                      ;DMA0転送停止
SET1    DMAMK0                    ;INTDM0割り込み処理禁止
RETI

```


C言語の処理も、アセンブリ言語と同様な動作を行います。

```

/*****
INTDMA0割り込み処理
(INTDMA0使用, CSI受信用)
-----

再生中, EEPROMから1ページ分のデータを受信完了したときに発生する割り込み
処理です。使用する4個のEEPROMの選択, I2Sバス・インタフェースの動作開始と
動作停止を行います。

レジスタバンク1使用
*****/
__interrupt void fn_intdma(void)
{
    /* 通信動作の終了待ちの完了待ち */
    while(SSR02 & 0b0000000001000000)    {}

    /* 1ページ分のデータ受信完了 */
    P_CS |= 0b11110000;    /* ノン・アクティブ */
    ucEepromSeq = CEEPSEQ_INST;

    if(ucPlayMode==CPLAY){
        /* 再生中 */
        /*-----*/
        I2Sバス・インタフェース動作開始
        /*-----*/
        /* EEPROMの選択 */
        if((ucEepromCs & 0b10000000)==0b00000000){
            /* EEPROMが一周 */
            ushEepromAddress++;
            if(ushEepromAddress>=128000/256){
                /* I2Sバス・インタフェース動作停止 */
                P_LRCLK = 1; /* LRCLK 出力無効化(ハイ) */
                TT0L.0 = 1; /* TO00出力動作停止(トリガ・ビット) */
                ST0L.0 = 1; /* CSIO00出力動作停止(トリガ・ビット) */
                ST0L.2 = 1; /* CSIO10出力動作停止(トリガ・ビット) */
                ucPlayMode = CPLAY_END; /* 再生終了中 */
            }
            else{
                ucEepromCs = 0b11100000;
            }
        }
        else{
            ucEepromCs = (ucEepromCs|0b00001000) <<1;
        }
    }
    else if(ucPlayMode==CPLAY_START){
        /* 再生準備開始 */
        /* EEPROMからのリード開始 */
        ucPlayMode = CPLAY_SET; /* 再生準備 */
        P_CS = (P_CS & 0b00001111)|0b11010000; /* EEPROM選択 */
        bI2sMemoryPage = 0;
        CSIF10 = 0; /* INTCSI10割り込み要求クリア */
        CSIMK10 = 0; /* INTCSI10割り込み処理許可 */
        SS0L.2 = 1; /* CSI10動作開始(トリガ・ビット) */
        SIO10 = 0b00000011; /* 送信データ・セット(リード指定, INTCSI10発生) */
    }
    else if(ucPlayMode==CPLAY_SET){
        /* 再生準備中 */
        ucPlayMode = CPLAY; /* 再生中 */
        ucI2sAddress = &ucMemoryPage1[1]; /* 送信バッファの先頭を設定 */
        bI2sMemoryPage = 0; /* ページ1送信 */
        /* I2Sバス・インタフェース動作開始 */
        SS0L.0 = 1; /* CSIO0動作開始(トリガ・ビット) */
        P_LRCLK = 0; /* LRCLKの動作開始 */
        TS0L.0 = 1; /* TO00出力動作開始(トリガ・ビット) */
        SIO0 = ucMemoryPage1[0]; /* 送信データ・セット(INTCSI00発生) */
    }
    DST1 = 0; /* DMA1転送停止 */
    DST0 = 0; /* DMA0転送停止 */
    DMAMK0 = 1; /* INTDMA0割り込み処理禁止 */
}

```

4.9 INTDMA1割り込み処理

アセンブリ言語のINTDMA1割り込み処理では、次の動作を行います。

レジスタ・バンクを切り替えます。

CSI10の通信動作が終了するまでウェイトします。

EEPROMの選択を行います。EEPROM全ページヘデータのライト動作が終了した場合、I²Sバス・インタフェースの動作を停止し、動作状態を録音停止とします。

DMAの動作を停止します。

```

;*****
;
;   INTDMA1割り込み処理
;   (INTDMA1使用, CSI送信用)
;
;-----
;
;   録音中, EEPROMへ1ページ分のデータを送信完了したときに発生する割り込み
;   処理です。使用する4個のEEPROMの選択とI2Sバス・インタフェースの動作停止
;   を行います。
;
;   レジスタバンク1使用
;*****
IINTDMA1:
-----
SEL      RB1
;通信動作の終了待ち
HDMA1100:
MOV      A,      SSR02L          ;通信動作完了?
BT       A,6,     $HDMA1100     ; No,

;1ページ分のデータ送信完了
OR       P_CS,   #11110000B     ;1ページ分のデータ送信完了
MOV      REEPSEQ, #CEEPSEQ_RESET

;EEPROMの選択
MOV      A,      RP_CS
SET1     A,3
ROL      A,      1
BC       $HDMA1300             ;EEPROMは一周した? No,
INCW    REEPADDR
MOVW    AX,     REEPADDR
CMPW    AX,     #128000/256     ;EEPROMの終端?
BNC     $HDMA1500             ; Yes,
MOV      A,      #11100000B
HDMA1300:
AND      A,      #11110000B
MOV      RP_CS,  A
BR      HDMA1RET
HDMA1500:
;I2Sバス・インタフェース動作停止
SET1     P_LRCLK              ;LRCLK 出力無効化(ハイ)
SET1     TT0L.0              ;TO00出力動作停止(トリガ・ビット)
SET1     ST0L.0              ;CSIO00出力動作停止(トリガ・ビット)
SET1     ST0L.2              ;CSIO10出力動作停止(トリガ・ビット)
MOV      RPLAYMOD, #CREC_END  ;録音停止
HDMA1RET:
CLR1     DST1                ;DMA1転送停止
CLR1     DST0                ;DMA0転送停止
SET1     DMAMK1              ;INTDM1割り込み処理禁止
RETI

```

C言語の処理も、アセンブリ言語と同様な動作を行います。

```

/*****
INTDMA1割り込み処理
(INTDMA1使用, CSI送信用)
-----
録音中, EEPROMへ1ページ分のデータを送信完了したときに発生する割り込み
処理です。使用する4個のEEPROMの選択とI2Sバス・インタフェースの動作停止
を行います。

レジスタバンク1使用
*****/
__interrupt void fn_intdma1(void)
{
    /* 通信動作の終了待ちの完了待ち */
    while(SSR02 & 0b0000000001000000)    {}

    /* 1ページ分のデータ送信完了 */
    P_CS |= 0b11110000;    /* 1ページ分のデータ送信完了 */
    ucEepromSeq = CEEPSEQ_RESET;

    /* EEPROMの選択 */
    if((ucEepromCs & 0b10000000)==0b00000000){
        ushEepromAddress++;
        if(ushEepromAddress>=128000/256){
            /* I2Sバス・インタフェース動作停止 */
            P_LRCLK = 1;    /* LRCLK 出力無効化(ハイ) */
            TT0L.0 = 1;    /* TO00出力動作停止(トリガ・ビット) */
            ST0L.0 = 1;    /* CSIO00出力動作停止(トリガ・ビット) */
            ST0L.2 = 1;    /* CSIO10出力動作停止(トリガ・ビット) */
            ucPlayMode = CPLAY_END;    /* 再生停止 */
        }
        else{
            ucEepromCs = 0b11100000;
        }
    }
    else{
        ucEepromCs = (ucEepromCs|0b00001000) <<1;
    }
    DST1 = 0;    /* DMA1転送停止 */
    DST0 = 0;    /* DMA0転送停止 */
    DMAMK1 = 1;    /* INTDM1割り込み処理禁止 */
}

```

4.10 I²Cバス・インタフェース・ライト処理

アセンブリ言語のオーディオCODEC内のレジスタ設定用I²Cバス・インタフェース・ライト処理では、次の動作を行います。

スタート・コンディションを発行します。

スレーブ・アドレスを送信します。

ライトするオーディオCODEC内のレジスタ・アドレスを送信します。

ライトする設定データを送信します。

ストップ・コンディションを発行します。

ウェイトが必要な場合、ウェイトを行います。

送信動作を終了するかのチェックを行います。送信動作を終了しない場合は次のデータがありますので の処理に戻ります。

```

;*****
;
;   オーディオCODEC内のレジスタ設定用I2Cバス・インタフェース・ライト処理
;
;-----
;
;   I2Cバス・インタフェースIIC0の機能を使用し、オーディオCODEC内のレジスタ
;   の設定データの書き込みを行います。
;-----
;
;   [I N]   ES      :レジスタ設定値ROMテーブルの先頭アドレス上位4ビット
;           HL      :レジスタ設定値ROMテーブルの先頭アドレス下位8ビット
;           DE      :レジスタ設定値ROMテーブルの最終アドレス下位8ビット
;   [OUT]   -
;
;*****
SI2CWRITE:
SET1      IICE0                      ;動作許可
JI2CW100:
;スタート・コンディション発行
SET1      STT0

;スレーブ・アドレス送信
BF        STD0,    $$                ;アドレス送信期間? No,
CLR1     IICIF0
MOV       IIC0,    #00110100B        ;アドレス送信開始(ライト指定)
BF        IICIF0,  $$                ;送信完了? No,
BT        ACKD0,   $JI2CW200         ;ACK検出? Yes,
SET1     SPT0      ; No, ストップ・コンディション発行
BR        $JI2CW100                  ;再送信

JI2CW200:
;ライト・アドレス送信
MOV       A,        ES:[HL]          ;設定データ取得
CLR1     IICIF0     ;INTIIC10割り込み要求クリア
MOV       IIC0,    A                ;データ送信開始
BF        IICIF0,  $$                ;送信完了? No,
BT        ACKD0,   $JI2CW300         ;ACK検出? Yes,
SET1     SPT0      ; No, ストップ・コンディション発行
BR        $JI2CW100                  ;再送信

JI2CW300:
;ライト・データ送信
INCW     HL          ;参照位置更新
MOV       A,        ES:[HL]          ;設定値取得
CLR1     IICIF0     ;INTIIC10割り込み要求クリア
MOV       IIC0,    A                ;データ送信開始
BF        IICIF0,  $$                ;送信完了? No,
BT        ACKD0,   $JI2CW400         ;ACK検出? Yes,
SET1     SPT0      ; No, ストップ・コンディション発行
DECW     HL          ;再送信
BR        $JI2CW100

JI2CW400:
;ストップ・コンディション発行
SET1     SPT0

;ウェイト制御
INCW     HL          ;参照位置更新
MOV       A,        ES:[HL]          ;ウェイト情報取得
CMP0     A          ;ウェイト必要?
BZ       $JI2CW600  ; No,
MOV       X,        #195             ;5ms
MULU     X          ;Aレジスタ*5ms
DECW     AX         ;AXレジスタ:タイマ・カウント値
MOVW     TDR01,    AX                ;インターバル設定
CLR1     TMIF01     ;INTTM01割り込み要求クリア
SET1     TSOL.1     ;タイマ動作開始
BF       TMIF01,   $$                ;インターバル経過? No,
SET1     TTOL.1     ;タイマ動作停止

JI2CW600:
INCW     HL          ;参照位置更新
MOVW     AX,        HL
CMPW     AX,        DE                ;レジスタ設定終了?
BC       $JI2CW100                  ; No,

JI2CWRET:
RET

```

C言語の処理も、アセンブリ言語と同様な動作を行います。

```

/*****
オーディオCODEC内のレジスタ設定用I2Cバス・インタフェース・ライト処理
-----

I2Cバス・インタフェースIIC0の機能を使用し、オーディオCODEC内のレジスタ
の設定データの書き込みを行います。
-----

[I N] *addr    : レジスタ設定値ROMテーブルの先頭アドレス上位4ビット
      size     : ROMテーブルのサイズ
[OUT] -

*****/
void fn_I2cWrite(unsigned char *addr, unsigned char size)
{
    register unsigned char cnt;

    IICE0 = 1;    /* 動作許可 */

    for(cnt=0; cnt<=size/3; cnt++){
        /* スタート・コンディション発行 */
        STT0 = 1;

        /* スレーブ・アドレス送信 */
        while(!STD0) {} /* アドレス送信期間? No, */
        IICIF0 = 0;
        IIC0 = 0b00110100; /* アドレス送信開始(ライト指定) */
        while(!IICIF0) {} /* 送信完了? No, */
        if(ACKD0){
            /* ライト・アドレス送信 */
            IICIF0 = 0; /* INTIIC10割り込み要求クリア */
            IIC0 = *addr; /* データ送信開始 */
            while(!IICIF0) {} /* 送信完了? No, */

            if(ACKD0){
                /* ライト・データ送信 */
                addr++; /* 参照位置更新 */
                IICIF0 = 0; /* INTIIC10割り込み要求クリア */
                IIC0 = *addr; /* データ送信開始 */
                while(!IICIF0) {} /* 送信完了? No, */

                if(ACKD0){
                    /* ストップ・コンディション発行 */
                    SPT0 = 1;

                    /* ウェイト制御 */
                    addr++; /* 参照位置更新 */
                    if(*addr!=0){
                        TDR01 = 195*(*addr)-1; /* インターバル設定(5ms*addr) */
                        TMIF01 = 0; /* INTTM01割り込み要求クリア */
                        TSOL.1 = 1; /* タイマ動作開始 */
                        while(!TMIF01) {} /* インターバル経過? No, */
                        TTOL.1 = 1; /* タイマ動作停止 */
                    }
                    addr++; /* 参照位置更新 */
                }
            }
            else{
                SPT0 = 1; /* ストップ・コンディション発行 */
                cnt--; /* 再送信 */
                addr--;
            }
        }
        else{
            SPT0 = 1; /* ストップ・コンディション発行 */
            cnt--; /* 再送信 */
        }
    }
    else{
        SPT0 = 1; /* ストップ・コンディション発行 */
        cnt--; /* 再送信 */
    }
}
}

```

第5章 関連資料

資料名		和文 / 英文
78K0R/KE3 ユーザーズ・マニュアル		PDF
78K0R/KF3 ユーザーズ・マニュアル		PDF
78K0R/KG3 ユーザーズ・マニュアル		PDF
78K0R/KH3 ユーザーズ・マニュアル		PDF
78K0R/KJ3 ユーザーズ・マニュアル		PDF
78K0Rマイクロコントローラ 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
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、78K0R/KG3マイクロコントローラのソース・プログラムを次に示します。

main.asm (アセンブリ言語版)

```
*****
;
;
;   NEC Electronics       78K0R/KG3 シリーズ
;
;*****
;
;   78K0R/KG3 シリーズ
;*****
;
;   オーディオ CODEC との I2S バス・インタフェース
;*****
;
; 【概要】
;
; このサンプル・プログラムは、I2S バス・インタフェースを応用し、オーディオ CODEC と
; のオーディオ・データの送受信を行うものです。I2S バス・インタフェースの L チャンネル
; か R チャンネルを選択するための LR クロック (LRCLK) をタイマ・アレイ・ユニット (TAU)
; のインターバル・タイム・モードを使用し、T000 より出力します。データを送受信する
; ためのクロック (BCLK) の出力、データの受信 (SDOUT) およびデータの送信 (SDIN)
; をシリアル・アレイ・ユニット 0 (SAU0) のチャンネル 0 の CS100 を使用します。
; OKI セミコンダクタ製のオーディオ CODEC ML2612 を使用し、オーディオ CODEC 内のレジスタ設定
; に I2C バス・インタフェースを使用し、オーディオ・データには I2S バス・インタフェー
; スを使用します。録音キー押下時はオーディオ CODEC によるマイク入力した音声データ
; を受信し、EEPROM ヘデータを保存します。また、再生キー押下時は録音したデータを
; オーディオ CODEC ヘ送信し、スピーカ出力を行います。EEPROM との SPI インタフェースは
; シリアル・アレイ・ユニット 0 (SAU0) のチャンネル 2 の CS110 を使用し、DMA コントローラ
; により CS1 送信 / 受信を続けて行います。録音中、再生中の状態は、動作中インジケー
; タ用 LED を P72 で出力し、表示させます。
;
; サンプル・プログラムを実現させる機能として以下のものを使用します。
;
;   オーディオ CODEC 内のレジスタ設定用 I2C バス・インタフェース
;     ・シリアル・インタフェース IIC0
;
;   オーディオ CODEC とのオーディオ・データ用 I2S バス・インタフェース
;     ・LRCLK としてタイマ・アレイ・ユニット (TAU) の T000 出力
;     ・データの送受信にシリアル・アレイ・ユニット 0 (SAU0) のチャンネル 0 の CS100
;
;   EEPROM との SPI インタフェース
;     ・EEPROM を 4 個使用のためチップ・セレクト端子として P74-P77
;     ・シリアル・アレイ・ユニット 0 (SAU0) のチャンネル 2 の CS110
;     ・送信 / 受信を続けて行うため、DMA コントローラの DMA0, DMA1
;
;   ウェイト用タイマ
;     ・5ms 以上のプログラム内ウェイト用としてタイマ・アレイ・ユニット (TAU) の
;       チャンネル 1 のインターバル・タイム・モード
;
;   録音 / 再生キー
;     ・キー割り込み入力端子 KR0, KR1
;
;   動作中インジケータ (LED) 出力
;     ・動作中インジケータ用 LED 出力として P72 で出力
;
;<使用する周辺の初期設定の主な設定内容>
;   割り込みの禁止
;   CPU / 周辺ハードウェア・クロック周波数を X1 発振クロック (20MHz 使用) に設定
;   ポートの設定
;   オーディオ CODEC への設定
;     ・オーディオ CODEC へのクロック供給
;     ・プログラム内ウェイト用のタイマ設定
```



```

DW      I INTDMA0           ; (1A) INTDMA0
DW      I INTDMA1           ; (1C) INTDMA1
DW      I INTCSI00          ; (1E) INTST0/INTCSI00
DW      RESET_START        ; (20) INTSR0/INTCSI01
DW      RESET_START        ; (22) INTSRE0
DW      I INTCSI10         ; (24) INTST1/INTCSI10/INTIIC10
DW      RESET_START        ; (26) INTSR1
DW      RESET_START        ; (28) INTSRE1
DW      RESET_START        ; (2A) INTIIC0
DW      I INTTMO0           ; (2C) INTTMO0
DW      RESET_START        ; (2E) INTTMO1
DW      RESET_START        ; (30) INTTMO2
DW      RESET_START        ; (32) INTTMO3
DW      RESET_START        ; (34) INTAD
DW      RESET_START        ; (36) INTRTC
DW      RESET_START        ; (38) INTRTC1
DW      RESET_START        ; (3A) INTKR
DW      RESET_START        ; (3C) INTST2/INTCSI20/INTIIC20
DW      RESET_START        ; (3E) INTSR2
DW      RESET_START        ; (40) INTSRE2
DW      RESET_START        ; (42) INTTMO4
DW      RESET_START        ; (44) INTTMO5
DW      RESET_START        ; (46) INTTMO6
DW      RESET_START        ; (48) INTTMO7
DW      RESET_START        ; (4A) INTP6
DW      RESET_START        ; (4C) INTP7
DW      RESET_START        ; (4E) INTP8
DW      RESET_START        ; (50) INTP9
DW      RESET_START        ; (52) INTP10
DW      RESET_START        ; (54) INTP11

TBRK   CSEG   AT      00007EH
       DW     RESET_START ; (7E) BRK

;=====
;
;   スタック領域の確保
;
;=====
DSTK   DSEG   AT      OFFEC0H
STACKEND:
       DS     20H          ;スタック領域を 32 バイト確保
STACKTOP:                          ;スタック領域の先頭アドレス = FFE0H

;=====
;
;   ポート定義
;
;=====
;オーディオ CODEC リセット
P_RESETB EQU P7.3          ;リセット
PM_RESETB EQU PM7.3        ;リセット出力

;オーディオ CODEC 内のレジスタ設定用 I2C インタフェース
P_SCL EQU P6.0             ;SCL0 ラッチ
PM_SCL EQU PM6.0           ;SCL0 出力
P_SDA EQU P6.1             ;SDA0 ラッチ
PM_SDA EQU PM6.1           ;SDA0 入出力

;オーディオ CODEC I2S バス・インタフェース
P_LRCLK EQU P0.1           ;LRCLK
PM_LRCLK EQU PM0.1         ;LRCLK 出力
P_SCK00 EQU P1.0           ;SCK00
PM_SCK00 EQU PM1.0         ;SCK00 出力
PM_S100 EQU PM1.1          ;S100 入力
P_S000 EQU P1.2            ;S000
PM_S000 EQU PM1.2          ;S000 出力

;キー入力 (2 本使用)
P_KEY EQU P7                ;KR(P70-P71 を使用)
PM_KEY EQU PM7              ;KR 入力(PM70-PM71 を使用)
P_PLAYKY EQU P7.0           ;再生キー
PM_PLAYKY EQU PM7.0         ;再生キー入力

```

```

P_RECKY EQU P7.1 ;録音キー
PM_RECKY EQU PM7.1 ;録音キー入力

;動作中インジケータ用 LED 出力 (1 本使用)
P_LED EQU P7.2 ;動作中インジケータ(Low アクティブ)
PM_LED EQU PM7.2 ;動作中インジケータ(LED)出力
; 動作中に点灯させ、キー入力が可能なときに消灯させます。

;EEPROM SPI インタフェース (4 個使用)
P_CS EQU P7 ;チップ・セレクト CS(P73-P77 を使用)
PM_CS EQU PM7 ;チップ・セレクト CS 出力(PM73-PM77 を使用)
P_CS3 EQU P7.7 ;CS3
PM_CS3 EQU PM7.7 ;CS3 出力
P_CS2 EQU P7.6 ;CS2
PM_CS2 EQU PM7.6 ;CS2 出力
P_CS1 EQU P7.5 ;CS1
PM_CS1 EQU PM7.5 ;CS1 出力
P_CS0 EQU P7.4 ;CS0
PM_CS0 EQU PM7.4 ;CS0 出力
P_SCK10 EQU P0.4 ;SCK10
PM_SCK10 EQU PM0.4 ;SCK10 出力
PM_S110 EQU PM0.3 ;S110 入力
P_S010 EQU P0.2 ;S010
PM_S010 EQU PM0.2 ;S010 出力

;=====
;
;
; RAM の定義
;
;=====
DPLAY DSEG SADDR
;全体の動作
RPLAYMOD: DS 1 ;動作状態
CRESET EQU 0 ;リセット
CSTOP EQU 1 ;停止中
CREC EQU 2 ;録音中
CREC_END EQU 3 ;録音終了中
CPLAY_START EQU 4 ;再生準備開始
CPLAY_SET EQU 5 ;再生準備中
CPLAY EQU 6 ;再生中
CPLAY_END EQU 7 ;再生終了中

;オーディオ CODEC との I2S バス・インタフェース動作関連
RCSI00CNT: DS 1 ;I2S データの 1 バイト送受信カウンタ
RPLAYINFO: DS 1 ;プレイ情報
FI2SPAGE EQU RPLAYINFO.7 ; I2S バスの送受信中のページ: ページ 1(0) / ページ 2(1)

;EEPROM との SPI インタフェース動作関連
RP_CS: DS 1 ;EEPROM のチップ・セレクト CS 設定値
REEPSEQ: DS 1 ;EEPROM の転送シーケンス
CEEPSEQ_RESET EQU 0 ;リセット状態
CEEPSEQ_WREN EQU 1 ;ライトイネーブル送信
CEEPSEQ_INST EQU 2 ;命令バイト送信
CEEPSEQ_ADDRH EQU 3 ;24 ビットアドレス上位 8 ビット部分送信
CEEPSEQ_ADDRM EQU 4 ;24 ビットアドレス中位 8 ビット部分送信
CEEPSEQ_ADDRL EQU 5 ;24 ビットアドレス下位 8 ビット部分送信
CEEPSEQ_DATA EQU 6 ;DMA によるデータ送信 / 受信

DPLAYP DSEG SADDRP
;オーディオ CODEC との I2S バス・インタフェース動作関連
RI2SADDR: DS 2 ;I2S 送受信データの保存先アドレス

;EEPROM との SPI インタフェース動作関連
REEPADDR: DS 2 ;EEPROM のリード/ライト・アドレス(REEPADDR*100H)

DMEM DSEG UNITP
;内蔵 RAM のメモリ領域
RRECMEM1: ;ページ 1
DS 3*2*42 ; LRCLK の 42 クロック分
RRECMEM1E: ; 最終アドレス+1
RRECMEM2: ;ページ 2
DS 3*2*42 ; LRCLK の 42 クロック分
RRECMEM2E: ; 最終アドレス+1

```

```

=====
;
; ROM の定義
;
=====
CREGACC CSEG UNITP
-----
;
; OKI セミコンダクタ製のオーディオ CODEC ML2612 の状態を遷移させるためのオーディオ
; CODEC 内のレジスタ設定を行うための設定値 ROM テーブル群となります。
; I2C バス・インタフェース・ライト処理で使用します。
;
; 【例】
;
; DB 0E5H, 00000111B, 0 ;Trimming 1
; DB 0E9H, 00000001B, 0 ;Trimming 2
;
; -----
; | | | |
; | | | | +----- ウエイト情報(注)
; | | | | +----- ライト・データ
; +----- ライト・アドレス
;
; 注) データをライトした後のウエイト時間を示します。
; 5ms × (記述した数値) の時間でウエイトを行います。
; 0 の場合はウエイトなしとなります。
;
; このサンプル・プログラムの設定データは、録音・再生を行うための
; ほんの一例です。詳細はデータ・シート等を参考にしてください。
;

```

TSYSON:

```

-----
;
; パワーOFF システム ON
;
-----
;
; ; Trimming
;
-----
DB 0E5H, 00000111B, 0 ;Trimming 1
DB 0E9H, 00000001B, 0 ;Trimming 2
;
; -----
; ; Digital Block ON
;
-----
DB 001H, 003H, 0 ;Sampling Rate
DB 003H, 001H, 0 ;PLLNL
DB 005H, 000H, 0 ;PLLNH
DB 007H, 000H, 0 ;PLLML
DB 009H, 002H, 0 ;PLLMH
DB 00BH, 002H, 0 ;PLLDIV
DB 00FH, 00000001B, 0 ;CLK Input/Output control
DB 00DH, 00000001B, 0 ;Clock Enable
DB 00DH, 00000101B, 150/5 ;Clock Enable

```

TSYSONE:

TRECON:

```

-----
;
; システム ON 録音開始
;
-----
;
; ; Record Function Setting
;
; -----
; ; ボリューム設定
;
DB 033H, 03FH, 0 ;Mic Input Volume
DB 039H, 00000000B, 0 ;Mic Boost Volume
DB 049H, 00000001B, 0 ;AMP Volume Control Function Enable
DB 04BH, 00000000B, 0 ;Amplifier Volume Fader Control
DB 069H, 00011110B, 0 ;Volume Control Func Enable
DB 06BH, 00000010B, 0 ;Mixer & Volume Control
DB 06DH, 0FFH, 0 ;Record Digital Volume Control
;
; -----
; ; HPF1 ON
DB 067H, 00000001B, 0 ;Filter Func Enable
;
; -----
; ; HPF2 ON
;

```

```

;DB 067H, 0000000B, 0 ;Filter Func Enable
;DB 07FH, 0000000B, 0 ;HPF2 CutOff

;Programmable Equalizer 設定
;DB 06DH, 1111111B, 0 ;Record Digital Volume
;DB 067H, 0000001B, 0 ;Filter Func Enable
;DB 067H, 0000000B, 0 ;Filter Func Enable
;DB 07FH, 0000000B, 0 ;HPF2 CutOff
;DB 067H, 0000000B, 0 ;Filter Func Enable
;DB 075H, 0000000B, 0 ;EQ gain1 Band0
;DB 077H, 0000000B, 0 ;EQ gain1 Band1
;DB 079H, 0000000B, 0 ;EQ gain1 Band2
;DB 07BH, 0000000B, 0 ;EQ gain1 Band3
;DB 07DH, 0000000B, 0 ;EQ gain1 Band4
;DB 083H, 0000000B, 0 ;EQ Band0 Coef0H
;DB 085H, 0000000B, 0 ;EQ Band0 Coef1L
;DB 087H, 0000000B, 0 ;EQ Band0 Coef1H
;DB 089H, 0000000B, 0 ;EQ Band1 Coef0L
;DB 08BH, 0000000B, 0 ;EQ Band1 Coef0H
;DB 08DH, 0000000B, 0 ;EQ Band1 Coef1L
;DB 08FH, 0000000B, 0 ;EQ Band1 Coef1H
;DB 091H, 0000000B, 0 ;EQ Band2 Coef0L
;DB 093H, 0000000B, 0 ;EQ Band2 Coef0H
;DB 095H, 0000000B, 0 ;EQ Band2 Coef1L
;DB 097H, 0000000B, 0 ;EQ Band2 Coef1H
;DB 099H, 0000000B, 0 ;EQ Band3 Coef0L
;DB 09BH, 0000000B, 0 ;EQ Band3 Coef0H
;DB 09DH, 0000000B, 0 ;EQ Band3 Coef1L
;DB 09FH, 0000000B, 0 ;EQ Band3 Coef1H
;DB 0A1H, 0000000B, 0 ;EQ Band4 Coef0L
;DB 0A3H, 0000000B, 0 ;EQ Band4 Coef0H
;DB 0A5H, 0000000B, 0 ;EQ Band4 Coef1L
;DB 0A7H, 0000000B, 0 ;EQ Band4 Coef1H

;ALC 設定
DB 069H, 0000010B, 0 ;Volume Control Func Enable
;DB 0B1H, 0000000B, 0 ;ALC Mode
;DB 0B3H, 0000000B, 0 ;ALC Attack Time
;DB 0B5H, 0000000B, 0 ;ALC Decay Time
;DB 0B7H, 0000000B, 0 ;ALC Hold Time
;DB 0B9H, 0000000B, 0 ;ALC Target Level
;DB 0BBH, 0000000B, 0 ;ALC Max/Min Gain
;DB 0BDH, 0000000B, 0 ;Noise Gate Threshold
;DB 0BFH, 0000000B, 0 ;ALC Zero Cross Time OutPlayback Limiter Control Register
;DB 02FH, 0000000B, 0 ;ZC-CMP Power Management

;-----
; Record Analog Block ON
;-----
;VMID 生成回路 ON
DB 021H, 0000001B, 5/5 ;Reference Power Management
DB 021H, 0000010B, 5/5 ;Reference Power Management

;マイクバイアス 回路 ON
DB 021H, 0000110B, 0 ;Reference Power Management

;MIN 入力設定 (差動入力)
DB 05BH, 0000010B, 0 ;Mic IF Control

;Power Management Register
DB 023H, 0000100B, 0 ;Input Power Management
DB 023H, 0000101B, 0 ;Input Power Management

;-----
; Record start
;-----
DB 069H, 0001000B, 0 ;Volume Control Func Enable
DB 069H, 0001000B, 0 ;Volume Control Func Enable
DB 013H, 0000001B, 5/5 ;Record/Playback Run
DB 06BH, 00110010B, 0 ;Mixer & Volume Control
DB 069H, 0001100B, 0 ;Volume Control Func Enable
DB 069H, 0000100B, 0 ;Volume Control Func Enable

```

TRECONE:

TRECOFF:

```

-----
;
;   録音   システム ON
;
-----
;
; Record stop
;
-----
DB   06BH, 0000010B, 0      ;Mixer & Volume Control
DB   069H, 00001000B, 0      ;Volume Control Func Enable
DB   069H, 00011000B, 200/5 ;Volume Control Func Enable
DB   013H, 00000000B, 10/5  ;Record/Playback Run
;
-----
; Record Analog Block OFF
;
-----
DB   023H, 00001000B, 0      ;Input Power Management
DB   023H, 00000000B, 0      ;Input Power Management
DB   021H, 00000001B, 0      ;Reference Power Management
DB   021H, 00000000B, 0      ;Reference Power Management

```

TRECOFFE:

TPLAYON:

```

-----
;
;   システム ON   再生開始
;
-----
;
; Playback Function Setting
;
-----
;ボリューム設定
DB   03BH, 00000000B, 0      ;Speaker AMP Volume
DB   0C9H, 00000000B, 0      ;Boost Volume
DB   049H, 00000000B, 0      ;AMP Volume Control Function Enable
DB   04BH, 00000000B, 0      ;Amplifier Volume Fader Control
DB   069H, 00000000B, 0      ;Volume Control
DB   06BH, 00000010B, 0      ;Mixer & Volume
DB   071H, 11111111B, 0      ;Playback Digital Volume Control
;
-----
;Programable Equalizer 設定
;DB   067H, 00000000B, 0      ;Filter Func Enable
;DB   075H, 00000000B, 0      ;EQ gain1 Band0
;DB   077H, 00000000B, 0      ;EQ gain1 Band1
;DB   079H, 00000000B, 0      ;EQ gain1 Band2
;DB   07bH, 00000000B, 0      ;EQ gain1 Band3
;DB   07dH, 00000000B, 0      ;EQ gain1 Band4
;DB   083H, 00000000B, 0      ;EQ Band0 Coef0H
;DB   085H, 00000000B, 0      ;EQ Band0 Coef1L
;DB   087H, 00000000B, 0      ;EQ Band0 Coef1H
;DB   089H, 00000000B, 0      ;EQ Band1 Coef0L
;DB   08BH, 00000000B, 0      ;EQ Band1 Coef0H
;DB   08DH, 00000000B, 0      ;EQ Band1 Coef1L
;DB   08FH, 00000000B, 0      ;EQ Band1 Coef1H
;DB   091H, 00000000B, 0      ;EQ Band2 Coef0L
;DB   093H, 00000000B, 0      ;EQ Band2 Coef0H
;DB   095H, 00000000B, 0      ;EQ Band2 Coef1L
;DB   097H, 00000000B, 0      ;EQ Band2 Coef1H
;DB   099H, 00000000B, 0      ;EQ Band3 Coef0L
;DB   09BH, 00000000B, 0      ;EQ Band3 Coef0H
;DB   09DH, 00000000B, 0      ;EQ Band3 Coef1L
;DB   09FH, 00000000B, 0      ;EQ Band3 Coef1H
;DB   0A1H, 00000000B, 0      ;EQ Band4 Coef0L
;DB   0A3H, 00000000B, 0      ;EQ Band4 Coef0H
;DB   0A5H, 00000000B, 0      ;EQ Band4 Coef1L
;DB   0A7H, 00000000B, 0      ;EQ Band4 Coef1H
;
-----
;Playback Limiter 設定
DB   069H, 00011001B, 0      ;Volume Control Func Enable
;DB   0C1H, 00000000B, 0      ;Attack Time
;DB   0C3H, 00000000B, 0      ;Decay Time
;DB   0C5H, 00000000B, 0      ;Target Level
;DB   0C7H, 00000000B, 0      ;Max/Min Gain
;
-----

```

```

; Playback Analog Block ON
;-----
;VMID 生成回路 ON
DB 021H, 0000001B, 5/5 ;Reference Power Management
DB 021H, 0000010B, 5/5 ;Reference Power Management

;ミュート設定
DB 049H, 0000000B, 0 ;AMP Volume Control Function Enable
DB 049H, 0000010B, 0 ;AMP Volume Control Function Enable

;DAC 設定
DB 025H, 0000010B, 0 ;DAC Power
DB 055H, 0000010B, 0 ;Speaker AMP Output Control

;スピーカ・アンプ ON
DB 027H, 013H, 0 ;Power Management
DB 027H, 01FH, 0 ;Reference Power Management
DB 03BH, 033H, 0 ;Speaker AMP Volume
DB 04BH, 0000000B, 0 ;Amplifier Volume Fader Control
DB 049H, 0000011B, 0 ;AMP Volume Control Function Enable
DB 049H, 0000001B, 500/5 ;AMP Volume Control Function Enable

;-----
; Playback start
;-----
DB 069H, 00011101B, 0 ;Volume Control Func Enable
DB 069H, 00010101B, 0 ;Volume Control Func Enable
DB 013H, 0000010B, 5/5 ;Record/Playback Run
DB 06BH, 00110010B, 0 ;Mixer & Volume Control
DB 069H, 00011101B, 0 ;Volume Control Func Enable
DB 069H, 00001101B, 0 ;Volume Control Func Enable
TPLAYONE:

TPLAYOFF:
;-----
; 再生 システム ON
;-----
;-----
; Playback Stop
;-----
DB 06BH, 0000010B, 0 ;Mixer & Volume Control
DB 069H, 00001000B, 0 ;Volume Control Func Enable
DB 069H, 00011000B, 100/5 ;Volume Control Func Enable
DB 013H, 00000000B, 0 ;Record/Playback Run

;-----
; Playback Analog Block OFF
;-----
; スピーカアンプ BTL モード OFF
DB 04BH, 00000000B, 0 ;Amplifier Volume Fader Control
DB 049H, 0000001B, 0 ;AMP Volume Control Function Enable
DB 049H, 0000011B, 100/5 ;AMP Volume Control Function Enable
DB 027H, 013H, 0 ;AMP Power Management
DB 027H, 000H, 0 ;AMP Power Management
DB 025H, 00000000B, 0 ;DAC Power Management
DB 021H, 00000000B, 0 ;Reference Power Management
TPLAYOFFE:

*****
;
;
;
; 使用する周辺の初期設定
;
;
*****
XMAIN CSEG UNIT
RESET_START:

;-----
; 割り込み禁止
;-----
DI

```

```

;-----
; レジスタバンク設定
;-----
SEL    RBO
;-----
; スタック・ポインタの設定
;-----
MOVW   SP,    #LOWW STACKTOP    ;スタック・ポインタを設定
;-----
; クロック周波数の設定
;-----
; 20MHz の X1 発振回路で動作が行えるように設定します
;-----
MOV    CMC,    #01000001B        ;クロック動作モード
; |||||+----- AMPH: 10MHz < fMX 20MHz
; |||+++----- <000>
; ||+----- OSCSELS: P123/P124 端子を入力ポート
; |+----- <0>
; ++----- EXCLK/OSCSEL: X1 発振モード(20MHz)

MOV    CSC,    #01000000B        ;クロック動作ステータス制御
; |||||+----- HI0STOP: 高速内蔵発振回路動作
; ||++++----- <00000>
; |+----- XTSTOP: XT1 発振回路停止
; +----- MSTOP: X1 発振回路動作

MOV    OSMC,   #00000001B        ;動作スピード・モード
; |||||+----- FSEL: 10MHz を越える周波数で動作
; ++++++----- <00000>

MOV    OSTS,   #00000101B        ;発振安定時間: 2^15/ fX

HRST300:
NOP
BF     OSTC.2, $HRST300          ;クロック発振安定待ち

MOV    CKC,    #00011000B        ;クロック選択
; |||||+++----- MDIV2-0: CPU/周辺ハードウェア・クロック(fCLK)=fMX
; |||+----- <1>
; ||+----- MCMO: 高速システム・クロック ( fMX )
; |+----- <R>
; +----- CSS: メイン・システム・クロック (fMAIN)=fCLK
; +----- <R>
;-----
; ポート 0 の設定
;-----
MOV    P0,     #00011010B        ;P00,P02,P05-P06 の出力ラッチ Low, P01,P03-P04 の出力ラッチ HIGH
MOV    PM0,    #10000000B        ;P00-P06 を出力ポートに設定
;P01: オーディオ CODEC への LRCLK 出力
;P02: EEPROM への S010 出力
;P03: EEPROM への S110 入力
;P04: EEPROM への SCK10 出力
;P00,P05-P06: 未使用
;-----
; ポート 1 の設定
;-----
MOV    P1,     #00000000B        ;P10-P17 の出力ラッチ Low
MOV    PM1,    #00000000B        ;P10-P17 を出力ポートに設定
;P10: オーディオ CODEC への SCK00 出力
;P11: オーディオ CODEC への S100 入力
;P12: オーディオ CODEC への S000 出力
;P13-P17: 未使用
;-----
; ポート 2 の設定
;-----
MOV    P2,     #00000000B        ;P20-P27 の出力ラッチ Low
MOV    PM2,    #00000000B        ;P20-P27 を出力ポートに設定
;P20-P27: 未使用

```


ポート 3 の設定

```
MOV P3, #00000000B ;P30-P31 の出力ラッチ Low
MOV PM3, #11111100B ;P30-P31 を出力ポートに設定
;P30-P31: 未使用
```

ポート 4 の設定

```
MOV P4, #00000000B ;P40-P47 の出力ラッチ Low
MOV PM4, #00000000B ;P40-P47 を出力ポートに設定
;P40-P47: 未使用
```

ポート 5 の設定

```
MOV P5, #00000000B ;P50-P57 の出力ラッチ Low
MOV PM5, #00000000B ;P50-P57 を出力ポートに設定
;P50-P57: 未使用
```

ポート 6 の設定

```
MOV P6, #00000000B ;P60-P67 の出力ラッチ Low
MOV PM6, #00000000B ;P60-P67 を出力ポートに設定
;P60: オーディオ CODEC 内のレジスタ設定のための SCL0 出力
;P61: オーディオ CODEC 内のレジスタ設定のための SDA0 入出力
;P62-P67: 未使用
```

ポート 7 の設定

```
MOV P7, #00000000B ;P70-P77 の出力ラッチ Low
MOV PM7, #00000011B ;P70-P71 を入力ポート, P72-P77 を出力ポートに設定
;P70: 再生キー入力
;P71: 録音キー入力
;P72: 動作中インジケータ(LED)出力
;P73: オーディオ CODEC へのリセット出力
;P74: EEPROM への CS0 出力
;P75: EEPROM への CS1 出力
;P76: EEPROM への CS2 出力
;P77: EEPROM への CS3 出力
```

ポート 8 の設定

```
MOV P8, #00000000B ;P80-P87 の出力ラッチ Low
MOV PM8, #00000000B ;P80-P87 を出力ポートに設定
;P80-P87: 未使用
```

ポート 11 の設定

```
MOV P11, #00000000B ;P110-P111 の出力ラッチ Low
MOV PM11, #11111100B ;P110-P111 を出力ポートに設定
;P110-P111: 未使用
```

ポート 12 の設定

```
MOV P12, #00000000B ;P120 の出力ラッチ Low
MOV PM12, #11111110B ;P120 を出力ポートに設定
;P120: 未使用
```

ポート 13 の設定

```
MOV P13, #00000000B ;P130-P131 の出力ラッチ Low
MOV PM13, #11111100B ;P131 を出力ポートに設定
;P130-P131: 未使用
```

ポート 14 の設定

```
MOV P14, #00000000B ;P140-P145 の出力ラッチ Low
MOV PM14, #11000000B ;P140-P145 を出力ポートに設定
;P140-P145: 未使用
```

ポート 15 の設定

```
MOV P15, #00000000B ;P150-P157 の出力ラッチ Low
MOV PM15, #00000000B ;P150-P157 を出力ポートに設定
;P150-P157: 未使用
```

オーディオ CODEC 内のレジスタ設定

下記動作を行います。

- ・オーディオ CODEC へのクロック供給
- ・プログラム内ウエイト用のタイマ設定
- ・オーディオ CODEC へのリセット信号出力
- ・オーディオ CODEC 内のレジスタ設定用の I2C インタフェース設定

```
MOV RPLAYMOD,#CRESET ;リセット
```

;タイマ・アレイユニットのタイマ・クロック選択

```
SET1 !TAU0EN ;タイマ・アレイユニットの入力クロック供給
MOV !TPSOL, #10010000B ;動作クロック選択
;|||++++----- CK00: fCLK
;++++----- CK01: fCLK/2^9
```

; オーディオ CODEC へのクロック供給

; T002 出力(10MHz)

```
MOVW AX, #0000000000000000B ;動作モード設定
;|||||||||++++----- MD023-020: インターバル・タイマ・モード
;|||||||||++----- <00 固定>
;|||||||++----- CIS021-020: 未使用
;|||||+++----- STS022-020: ソフトウェア・トリガ・スタートのみ有効
;|||+----- MASTER02: 単体動作
;||+----- CSS02: CKS02 ビットで指定したマクロ・クロック MCK
;|++----- <00 固定>
;+----- CKS02: PRS レジスタで設定した動作クロック CK00
```

```
MOVW !TMR02, AX
```

```
MOVW TDR02,#0 ;インターバル設定 : 10MHz 出力(20MHz)
```

```
SET1 TOEOL.2 ;T002 出力
```

```
SET1 TSOL.2 ;動作開始(トリガ・ビット)
```

; プログラム内で使用するウエイト用

; TM01 使用

```
MOVW AX, #1000000000000000B ;動作モード設定
;|||||||||++++----- MD013-010: インターバル・タイマ・モード
;|||||||||++----- <00 固定>
;|||||||++----- CIS011-010: 未使用
;|||||+++----- STS012-010: ソフトウェア・トリガ・スタートのみ有効
;|||+----- MASTER01: 単体動作
;||+----- CSS01: CKS01 ビットで指定したマクロ・クロック MCK
;|++----- <00 固定>
;+----- CKS01: PRS レジスタで設定した動作クロック CK01
```

```
MOVW !TMR01, AX
```

;電源安定待ち

```
MOVW TDR01, #195*210/5-1 ;インターバル設定(210ms)
```

```
SET1 TSOL.1 ;タイマ動作開始
```

```
CLR1 TMIF01 ;INTTM00 割り込み要求クリア
```

```

BF      TMIF01, $$

;リセット
CLR1    P_RESETB
;ウエイト約 5us (50ns*5*Breg で算出)
MOV     B,      #5000/50/5
HRST550:
DEC     B                      ;1clk
BNZ     $HRST550                ;4clk(Z=0)
SET1    P_RESETB

;電源安定待ち
MOVW    TDR01, #195*160/5-1     ;インターバル設定(160ms)
SET1    TSOL.1                 ;タイマ動作開始
CLR1    TMIF01                  ;INTTMOO 割り込み要求クリア
BF      TMIF01, $$
SET1    TTOL.1                  ;タイマ動作停止

;-----
; オーディオ CODEC 内のレジスタ設定値の送信
;-----
; IIC0 使用
;-----
SET1    !IIC0EN                 ;シリアル・インタフェース IIC0 の入力クロック供給

CLR1    IICE0                   ;動作停止

CLR1    P_SCL                   ;SCL0 ラッチ
CLR1    PM_SCL                  ;SCL0 端子
CLR1    P_SDA                   ;SDA0 端子
CLR1    PM_SDA                  ;SDA0 端子

MOV     IICX0, #00000000B       ;転送クロック選択
; |||||+----- CLX0
; ++++++----- <0000000>

MOV     IICCL0, #00001110B
; |||||++----- CL01-00(+CLX0): fCLK/96(ファースト・モード)
; |||||+----- DFCO: デジタル・フィルタ・オン
; ||||+----- SMC0: ファースト・モードで動作
; ||+----- DADO: <R>SDA0 端子のレベル検出
; |+----- CLDO: <R>SCL0 端子のレベル検出
; ++----- <00>

MOV     IICF0, #00000011B       ;通信予約禁止
; |||||+----- IICRSV: 通信予約機能禁止
; |||||+----- STCEN: 初期スタート許可
; ||+++----- <0000 固定>
; |+----- IICBSY: <R>IIC バス状態フラグ
; +----- STCF: <R>STT0 クリアフラグ

MOV     IICCO, #00001000B       ;マスタ時の初期設定
; |||||+----- SPT0: ストップ・コンディション・トリガ
; |||||+----- STT0: スタート・コンディション・トリガ
; ||||+----- ACKEO: アクノリッジ制御
; |||+----- WTIMO: ウエイトおよび割込要求発生時の制御:9 クロック
; ||+----- SPIEO: ストップコンディション検出による割込要求発生禁止
; |+----- WRELO: ウエイト解除しない
; +----- LRELO: 通信退避:通常動作
; +----- IICE0: I2C の動作許可

SET1    SPT0                    ;ストップ・コンディション設定

;オーディオ CODEC 内のレジスタ設定: システム ON 設定
MOV     ES,      #HIGHW TSYSON   ;レジスタ設定値 ROM テーブルの先頭アドレス上位 4 ビット
MOVW    HL,      #LOWW TSYSON    ;下位 16 ビット
MOVW    DE,      #LOWW TSYSONE   ;レジスタ設定値 ROM テーブルの最終アドレス下位 8 ビット
CALL    !!S12CWRITE              ;I2C ライト処理

MOV     RPLAYMOD, #CSTOP         ;停止

```

オーディオ CODEC のオーディオ・データ用 I2S バス・インタフェース設定

下記動作を行います。

- ・ LRCLK 出力用に T000 出力と INTTM00 割り込みの設定
- ・ データ送受信用に CS100 の設定

; LRCLK の出力設定

; T000 出力(16kHz)

```
MOVW AX, #000000000000000B ;動作モード設定
; |||||++++----- MD003-000: インターバル・タイマ・モード
; |||||++++----- <00 固定>
; |||||++----- CIS001-000: 未使用
; |||||+++----- STS002-000: ソフトウエア・トリガ・スタートのみ有効
; |||+----- MASTER00: 単体動作
; |||+----- CSS00: CKS00 ビットで指定したマクロ・クロック MCK
; |+----- <00 固定>
; +----- CKS00: PRS レジスタで設定した動作クロック CK00
```

```
MOVW !TMR00, AX
MOVW TDR00, #625-1 ;インターバル設定 : 16kHz 出力(32kHz)
```

```
CLR1 TMPR100 ;優先順位を最高位に設定
CLR1 TMPR000
```

; CS100 の設定

```
SET1 !SAU0EN ;シリアル・アレイ・ユニットの入力クロック供給
```

```
NOP ;ウエイト
NOP
NOP
NOP
```

```
MOV !SPS0L, #00000000B ;動作クロック選択 fCLK
; |||++++----- PRS003-00: fCLK
; +++----- PRS013-10: 未使用
```

```
MOVW SDR00, #(12-1) shl 9 ;ビット 15-7: 転送クロック設定(833kHz)
```

```
MOVW AX, #000000000100001B ;動作モード選択:CSI モード
; |||||++++----- MD000: バッファ空き割り込み
; |||||++++----- MD002-1: CSI モード
; |||||++++----- <100 固定>
; |||||++----- S1S000: 未使用
; |||||++----- <0 固定>
; |||||+----- STS00: ソフトウエア・トリガのみ有効(CSI 時は固定)
; ||++++----- <00000 固定>
; +----- CSS00: 転送クロックを CKS00 ビットで指定した動作クロック MCK の分周ク
; | ロックに設定
; +----- CKS00: 動作クロックを PRS レジスタで設定したプリスケラ出力クロック
; CK00 に設定
```

```
MOVW !SMR00, AX
```

;初期データ出力 (CS110 の設定時に同時に行う)

```
;MOVW AX, #0000000100000001B ;S0 端子&SCK 端子の初期出力設定
; ; |||||++++----- S00n-0: 送信で使用
; ; +----- CK0n-0: チャンネル n-0 のシリアル・クロック出力
;MOVW !S00, AX
```

```
SET1 SOE0L.0 ;出力許可
```

```
SET1 P_SCK00 ;SCK00 ラッチ:ハイ
CLR1 PM_SCK00 ;SCK00 端子出力設定
SET1 PM_S100 ;S100 端子入力設定
SET1 P_S000 ;S000 ラッチ:ハイ
CLR1 PM_S000 ;S000 端子出力設定
```



```

BR      HTMOORET

HTMOOPLAY:
;-----
; 再生中
;-----
; I2S バス・インタフェース動作開始
MOVW   HL,    R12SADDR      ; I2S 送信データの保存先アドレス
MOV     A,    [HL]         ; 送信データ(1バイト目)
SET1   SSOL.0             ; CSI100 動作開始(トリガ・ビット)
MOV     S100,  A          ; 送信データ・セット(INTCSI00 発生)

HTMOORET:
INCW   R12SADDR          ; I2S 送受信データの保存先アドレスの更新
RETI

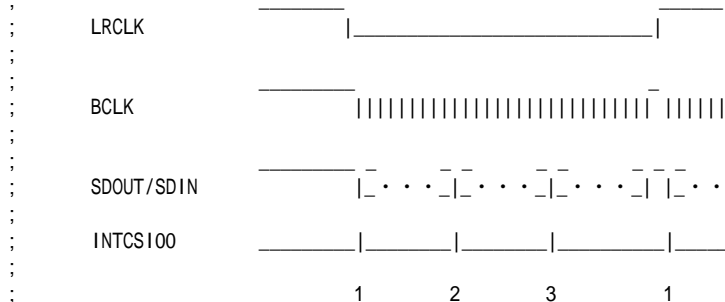
```

```

;*****
;
; INTCSI00 割り込み処理
; (INTCSI00 使用, CSI 送信/受信用)
;*****

```

オーディオ CODEC との I2S バス・インタフェースで、データの送受信を行います。録音中、CSI100 を使用し、受信データを保存します。通常はダミー・データをセットすることでデータの連続受信を行います。LRCLK との同期を行うため、3 バイト毎に受信のみを行い、連続受信の動作を停止させます。停止させた動作は、INTTMOO 割り込み処理で動作再開となり LRCLK と同期させることができます。受信したデータが 1 ページに達した場合は、EEPROM へのライト動作を開始します。再生中も録音中と同様に、CSI100 を使用し、データを連続送信しますが、LRCLK との同期を行うため、3 バイト毎に連続送信の動作を停止させます。停止させた動作は、INTTMOO 割り込み処理で動作再開となり LRCLK と同期させることができます。送信したデータが 1 ページに達した場合は、EEPROM から別の 1 ページ分のデータのリードを開始します。



レジスタバンク 1 使用

```

;*****
; INTCSI00:
SEL     RB1

INC     RCS100CNT          ; CSI100 受信カウンタ更新

CMP     RPLAYMOD, #CREC   ; 録音中?
BNZ    $H12SPLAY         ; No,

;-----
; 録音中
;-----
; ページの最終位置の確認
MOVW   AX,    R12SADDR      ; I2S 受信データの保存先アドレス
BT     F12SPAGE, $H12SR030
CMPW   AX,    #RRECMEM1E   ; ページ 1 の最終?
BC     $H12SR100          ; No,
MOVW   R12SADDR, #RRECMEM2 ; I2S 受信データの保存先アドレス: ページ 2
SET1   F12SPAGE
BR     H12SR050

H12SR030:
CMPW   AX,    #RRECMEM2E   ; ページ 2 の最終?
BC     $H12SR100          ; No,
MOVW   R12SADDR, #RRECMEM1 ; I2S 受信データの保存先アドレス: ページ 1

```

```

CLR1    F12SPAGE
HI2SR050:
;EEPROM へのライト開始
MOV     REEPSEQ,#CEEPSEQ_WREN      ;ライトイネーブル(WREN)送信
MOV     A,      P_CS                ;EEPROM CS 設定
AND     A,      #00001111B
OR      A,      RP_CS
MOV     P_CS,   A
CLR1    CS11F10                     ;INTCSI10 割り込み要求クリア
CLR1    CS1MK10                     ;INTCSI10 割り込み処理許可
SET1    SSOL.2                      ;CS110 動作開始(トリガ・ビット)
MOV     S1010, #00000110B          ;ライトイネーブル(WREN)命令(INTCSI10 発生)

HI2SR100:
;動作開始直後のバッファ空き割り込み
CMP     RCS100CNT,#1                ;受信動作開始直後のバッファ空き割り込み?
BNZ     $HI2SR200                   ; No,
MOV     S1000, #0FFH                ;ダミー・データのセット
BR      HI2SR200                    ; Yes, (不定なデータのため無視する)

HI2SR200:
;1 バイト目の受信
CMP     RCS100CNT,#2                ;1 バイト目の受信?
BNZ     $HI2SR300                   ; No,
MOV     A,      #0FFH                ;ダミー・データ
XCH     A,      S1000                ;ダミー・データのセットおよびデータの受信
BR      HI2SR500

HI2SR300:
;2 バイト目の受信
MOV     RCS100CNT,#0                ;CS100 受信カウンタ初期化
MOV     A,      S1000                ;データの受信のみ行う
;BR     HI2SR500

HI2SR500:
MOVW    HL,    R12SADDR              ;I2S 受信データの保存先アドレス
MOV     [HL],  A                    ;受信データ保存

HI2SR800:
INCW    R12SADDR                    ;I2S 受信データの保存先アドレスの更新
BR      HI2SR200

;-----
; 再生中
;-----

HI2SP100:
CMP     RPLAYMOD,#CPLAY              ;再生中?
BNZ     $HI2SR200                   ; No,

CMP     RCS100CNT,#3                ;LRCLK との同期タイミング?(3 バイト目の送信動作開始直後?)
BNC     $HI2SP100

MOVW    HL,    R12SADDR              ;I2S 送信データの保存先アドレス
MOV     A,      [HL]                ;送信データ(2 バイト目もしくは3 バイト目)
MOV     S1000,  A                    ;送信データ・セット(INTCSI100 発生)
INCW    R12SADDR                    ;I2S 送信データの保存先アドレスの更新
BR      HI2SP200

HI2SP100:
MOV     RCS100CNT,#0                ;CS100 送信カウンタ初期化

HI2SP200:
MOVW    AX,    R12SADDR              ;I2S 送信データの保存先アドレス
BT      F12SPAGE,$HI2SP300          ;ページ 2 使用中? Yes,
CMPW    AX,    #RRECMEM1E           ;ページ 1 の最終?
BC      $HI2SR200                    ; No,
MOVW    R12SADDR,#RRECMEM2          ;I2S 送信データの保存先アドレス: ページ 2
SET1    F12SPAGE
BR      HI2SP500

HI2SP300:
CMPW    AX,    #RRECMEM2E           ;ページ 2 の最終?
BC      $HI2SR200                    ; No,
MOVW    R12SADDR,#RRECMEM1          ;I2S 送信データの保存先アドレス: ページ 1
CLR1    F12SPAGE

HI2SP500:
;EEPROM へのリード開始
MOV     A,      P_CS                ;EEPROM 選択
AND     A,      #00001111B
OR      A,      RP_CS
MOV     P_CS,   A

```

```

MOV    REEPSEQ,#CEEPSEQ_INST      ;命令バイト送信
CLR1   CS1IF10                    ;INTCSI10 割り込み要求クリア
CLR1   CS1MK10                    ;INTCSI10 割り込み処理許可
SET1   SSOL.2                     ;CS110 動作開始(トリガ・ビット)
MOV    S1010, #00000011B         ;送信データ(リード指定)・セット(INTCSI10 発生)
HI2SRET:
    RETI

;*****
;
;
;   INTCSI10 割り込み処理
;   (INTCSI10 使用, CSI 送信 / 受信用)
;
;-----
;
;   録音中は, EEPROM への命令バイトの送信と 24 ビット・アドレスの送信を行い,
;   EEPROM へのデータ送信のタイミングで, DMA のチャンネル 1 を開始し, CS110 の
;   シングル送信にて続けて送信します。
;   再生中は, EEPROM への 24 ビット・アドレスの送信を行い, EEPROM からのデー
;   タ受信のタイミングで, DMA のチャンネル 0 とチャンネル 1 を開始し, CS110 のシ
;   グル受信にて続けて受信します。ただし, DMA のチャンネル 1 はダミー・デー
;   タのセット用となります。
;
;   レジスタバンク 1 使用
;
;*****
IINTCSI10:
    EI          ;多重割り込みの許可
    SEL        RB2

    INC        REEPSEQ          ;EEPROM の転送シーケンス更新

    CMP        RPLAYMOD,#CREC   ;録音中? No,(再生中)
    BZ         $HCSI11REC
    BR        HCSI1PLAY

HCSI11REC:
;-----
;   録音中
;-----
;命令バイト送信
CMP    REEPSEQ,#CEEPSEQ_INST    ;命令バイト送信タイミング?
BNZ   $HCSI11100                ; No,
OR    P_CS, #111110000B         ;ノン・アクティブ
NOP
MOV   A, P_CS                    ;EEPROM 選択
AND   A, #00001111B
OR    A, RP_CS
MOV   P_CS, A
MOV   A, #00000010B             ;ライト指定
BR    HCSI1650                  ;データ送信

HCSI11100:
;24 ビット・アドレス上位送信
CMP    REEPSEQ,#CEEPSEQ_ADDRH   ;24 ビット・アドレス上位送信タイミング?
BNZ   $HCSI1200                ; No,
MOV   A, REEPADDR+1
BR    HCSI1650                  ;データ送信

HCSI1200:
;24 ビット・アドレス中位送信
CMP    REEPSEQ,#CEEPSEQ_ADDRM   ;24 ビット・アドレス中位送信タイミング?
BNZ   $HCSI1300                ; No,
MOV   A, REEPADDR
BR    HCSI1650                  ;データ送信

HCSI1300:
;24 ビット・アドレス下位送信
CMP    REEPSEQ,#CEEPSEQ_ADDRL   ;24 ビット・アドレス下位送信タイミング?
BNZ   $HCSI1400                ; No,
MOV   A, #000H
BR    HCSI1650                  ;データ送信

HCSI1400:
;データ送信開始
;DMA 転送開始
BT    F12SPAGE,$HCSI1430       ;I2S はページ 2 を使用中?

```



```

BT      F12SPAGE,$HCS11P430      ; I2S はページ 2 を使用中?
MOVW   DRAO, #RRECMEM2          ; 受信バッファをページ 1 に設定
BR      HCS11P450

HCS11P430:
MOVW   DRAO, #RRECMEM1          ; 受信バッファをページ 2 に設定
HCS11P450:
MOVW   DBCO, #3*2*42           ; ページ・サイズ設定

; DMA 転送開始
SET1   DST1                     ; DMA1 転送許可 (CSI10 送信用)
SET1   DST0                     ; DMA0 転送許可 (CSI10 受信用)

SET1   DMAMK1                   ; INTDM1 割り込み処理禁止
CLR1   CS1IF10                  ; INTCSI10 割り込み要求クリア
SET1   CSIMK10                  ; INTCSI10 割り込み処理禁止

CLR1   DMAIF0                   ; INTDM0 割り込み要求クリア
CLR1   DMAMK0                   ; INTDM0 割り込み処理許可

SET1   SSOL.2                   ; CSI10 動作開始 (トリガ・ビット)
MOV    S1010, #0FFH            ; ダミー・データのセット (INTCSI100 発生)
HCS11RET:
RET1

```

```

;*****
;
;
;   INTDMAO 割り込み処理
;   (INTDMAO 使用, CSI 受信用)
;
;-----
;
;
;   再生中, EEPROM から 1 ページ分のデータを受信完了したときに発生する割り込み
;   処理です。使用する 4 個の EEPROM の選択, I2S バス・インタフェースの動作開始と
;   動作停止を行います。
;
;
;   レジスタバンク 1 使用
;
;*****

```

```

I INTDMAO:
SEL    RB1

; 通信動作の終了待ち
HDMAO100:
MOV    A,      SSR02L           ; 通信動作完了?
BT     A.6,    $HDMAO100       ; No,

; 1 ページ分のデータ受信完了
OR     P_CS,   #11110000B      ; ノン・アクティブ
MOV    REEPSEQ,#CEEPSEQ_INST

CMP    RPLAYMOD,#CPLAY         ; 再生中?
BZ     $HDMAO500               ; Yes,
CMP    RPLAYMOD,#CPLAY_SET     ; 再生準備?
BZ     $HDMAO300               ; Yes,
CMP    RPLAYMOD,#CPLAY_START   ; 再生準備開始?
BNZ    $HDMAORET               ; No,

; EEPROM からのリード開始
MOV    RPLAYMOD,#CPLAY_SET     ; 再生準備
MOV    A,      P_CS            ; EEPROM 選択
AND    A,      #00001111B
OR     A,      #11010000B
MOV    P_CS,   A
CLR1   F12SPAGE
CLR1   CS1IF10                  ; INTCSI10 割り込み要求クリア
CLR1   CSIMK10                  ; INTCSI10 割り込み処理許可
SET1   SSOL.2                   ; CSI10 動作開始 (トリガ・ビット)
MOV    S1010, #0000011B       ; 送信データ (リード指定)・セット (INTCSI100 発生)
BR     HDMAORET

;-----
;   I2S バス・インタフェース動作開始
;-----

```

```

HDMA0300:
MOV    RPLAYMOD,#CPLAY           ;再生中
MOVW   R12SADDR,#RRECMEM1+1     ;送信バッファの先頭を設定
CLR1   F12SPAGE
MOV    A,    !RRECMEM1
;I2S バス・インタフェース動作開始
SET1   SSOL.0                    ;CS100 動作開始(トリガ・ビット)
CLR1   P_LRCLK                   ;LRCLK の動作開始
SET1   TSOL.0                    ;T000 出力動作開始(トリガ・ビット)
MOV    S1000, A                  ;送信データ・セット(INTCS100 発生)

BR     HDMA0RET

HDMA0500:
;EEPROM の選択
MOV    A,    RP_CS
SET1   A.3
ROL    A,    1
BC     $HDMA0700                 ;EEPROM は一周した? No,
IN    REEPADDR
MOVW   AX,   REEPADDR
CMPW   AX,   #128000/256         ;EEPROM の終端?
BNC    $HDMA0800                ; Yes,
MOV    A,    #11110000B

HDMA0700:
AND    A,    #11110000B
MOV    RP_CS, A
BR     HDMA0RET

HDMA0800:
;I2S バス・インタフェース動作停止
SET1   P_LRCLK                   ;LRCLK 出力無効化(ハイ)
SET1   TTOL.0                   ;T000 出力動作停止(トリガ・ビット)
SET1   STOL.0                   ;CS1000 出力動作停止(トリガ・ビット)
SET1   STOL.2                   ;CS1010 出力動作停止(トリガ・ビット)
MOV    RPLAYMOD,#CPLAY_END      ;再生終了中

HDMA0RET:
CLR1   DST1                      ;DMA1 転送停止
CLR1   DST0                      ;DMA0 転送停止
SET1   DMAMKO                    ;INTDMO 割り込み処理禁止
RET

```

```

;*****
;
;
;   INTDMA1 割り込み処理
;   (INTDMA1 使用, CSI 送信用)
;
;-----
;
;   録音中, EEPROM へ 1 ページ分のデータを送信完了したときに発生する割り込み
;   処理です。使用する 4 個の EEPROM の選択と I2S バス・インタフェースの動作停止
;   を行います。
;
;   レジスタバンク 1 使用
;*****

```

```

| INTDMA1:
SEL    RB1

;通信動作の終了待ち
HDMA1100:
MOV    A,    SSR02L              ;通信動作完了?
BT     A.6,   $HDMA1100         ; No,

;1 ページ分のデータ送信完了
OR     P_CS,  #11110000B        ;1 ページ分のデータ送信完了
MOV    REEPSEQ,#CEEPSEQ_RESET

;EEPROM の選択
MOV    A,    RP_CS
SET1   A.3
ROL    A,    1
BC     $HDMA1300                 ;EEPROM は一周した? No,
IN    REEPADDR
MOVW   AX,   REEPADDR

```



```

CMPW  AX, #128000/256 ;EEPROMの終端?
BNC   $HDMA1500 ; Yes,
MOV   A, #11100000B

HDMA1300:
AND   A, #11110000B
MOV   RP_CS, A
BR    HDMA1RET

HDMA1500:
;I2Sバス・インタフェース動作停止
SET1  P_LRCLK ;LRCLK出力無効化(ハイ)
SET1  TTOL.0 ;T000出力動作停止(トリガ・ビット)
SET1  STOL.0 ;CS1000出力動作停止(トリガ・ビット)
SET1  STOL.2 ;CS1010出力動作停止(トリガ・ビット)
MOV   RPLAYMOD,#CREC_END ;録音停止

HDMA1RET:
CLR1  DST1 ;DMA1転送停止
CLR1  DST0 ;DMA0転送停止
SET1  DMAMK1 ;INTDM1割り込み処理禁止
RET1

;*****
;
;
; オーディオ CODEC 内のレジスタ設定用 I2C バス・インタフェース・ライト処理
;
;-----
;
; I2C バス・インタフェース IIC0 の機能を使用し、オーディオ CODEC 内のレジスタ
; の設定データの書き込みを行います。
;
;-----
;
; [ I N ] ES :レジスタ設定値 ROM テーブルの先頭アドレス上位 4 ビット
;          HL :レジスタ設定値 ROM テーブルの先頭アドレス下位 8 ビット
;          DE :レジスタ設定値 ROM テーブルの最終アドレス下位 8 ビット
; [OUT] -
;
;*****
SI2CWWRITE:
SET1  IICE0 ;動作許可

JI2CW100:
;スタート・コンディション発行
SET1  STT0

;スレーブ・アドレス送信
BF    STD0, $$ ;アドレス送信期間? No,
CLR1  IICIF0
MOV   IIC0, #00110100B ;アドレス送信開始(ライト指定)
BF    IICIF0, $$ ;送信完了? No,
BT    ACKD0, $JI2CW200 ;ACK検出? Yes,
SET1  SPT0 ; No, ストップ・コンディション発行
BR    $JI2CW100 ;再送信

JI2CW200:
;ライト・アドレス送信
MOV   A, ES:[HL] ;設定データ取得
CLR1  IICIF0 ;INTIIC10 割り込み要求クリア
MOV   IIC0, A ;データ送信開始
BF    IICIF0, $$ ;送信完了? No,
BT    ACKD0, $JI2CW300 ;ACK検出? Yes,
SET1  SPT0 ; No, ストップ・コンディション発行
BR    $JI2CW100 ;再送信

JI2CW300:
;ライト・データ送信
INCW  HL ;参照位置更新
MOV   A, ES:[HL] ;設定値取得
CLR1  IICIF0 ;INTIIC10 割り込み要求クリア
MOV   IIC0, A ;データ送信開始
BF    IICIF0, $$ ;送信完了? No,
BT    ACKD0, $JI2CW400 ;ACK検出? Yes,
SET1  SPT0 ; No, ストップ・コンディション発行
DECW  HL ;再送信
BR    $JI2CW100

```

J12CW400:

;ストップ・コンディション発行
SET1 SPT0

;ウエイト制御

INCW	HL			;参照位置更新
MOV	A,	ES:	[HL]	;ウエイト情報取得
CMPO	A			;ウエイト必要?
BZ	\$J12CW600			; No,
MOV	X,	#195		;5ms
MULU	X			;A レジスタ*5ms
DECW	AX			;AX レジスタ:タイマ・カウント値
MOVW	TDR01,	AX		;インターバル設定
CLR1	TMIF01			;INTTM01 割り込み要求クリア
SET1	TSOL.1			;タイマ動作開始
BF	TMIF01,	\$\$;インターバル経過? No,
SET1	TTOL.1			;タイマ動作停止

J12CW600:

INCW	HL			;参照位置更新
MOVW	AX,	HL		
CMPW	AX,	DE		;レジスタ設定終了?
BC	\$J12CW100			; No,

J12CWRET:

RET

end

main.c (C言語版)

```

/*****
NEC Electronics    78K0R/KG3 シリーズ
*****
78K0R/KG3 シリーズ
*****
オーディオ CODEC との I2S バス・インタフェース
*****

```

【概要】

このサンプル・プログラムは、I2S バス・インタフェースを応用し、オーディオ CODEC とのオーディオ・データの送受信を行うものです。I2S バス・インタフェースの L チャネルか R チャネルを選択するための LR クロック (LRCLK) をタイマ・アレイ・ユニット (TAU) のインターバル・タイマ・モードを使用し、T000 より出力します。データを送受信するためのクロック (BCLK) の出力、データの受信 (SDOUT) およびデータの送信 (SDIN) をシリアル・アレイ・ユニット 0 (SAU0) のチャンネル 0 の CS100 を使用します。OKI セミコンダクタ製のオーディオ CODEC ML2612 を使用し、オーディオ CODEC 内のレジスタ設定に I2C バス・インタフェースを使用し、オーディオ・データには I2S バス・インタフェースを使用します。録音キー押下時はオーディオ CODEC によるマイク入力した音声データを受信し、EEPROM へデータを保存します。また、再生キー押下時は録音したデータをオーディオ CODEC へ送信し、スピーカ出力を行います。EEPROM との SPI インタフェースはシリアル・アレイ・ユニット 0 (SAU0) のチャンネル 2 の CS110 を使用し、DMA コントローラにより CSI 送信/受信を続けて行います。録音中、再生中の状態は、動作中インジケータ用 LED を P72 で出力し、表示させます。

サンプル・プログラムを実現させる機能として以下のものを使用します。

オーディオ CODEC 内のレジスタ設定用 I2C バス・インタフェース
 ・シリアル・インタフェース I1C0

オーディオ CODEC とのオーディオ・データ用 I2S バス・インタフェース
 ・LRCLK としてタイマ・アレイ・ユニット (TAU) の T000 出力
 ・データの送受信にシリアル・アレイ・ユニット 0 (SAU0) のチャンネル 0 の CS100

EEPROM との SPI インタフェース
 ・EEPROM を 4 個使用のためチップ・セレクト端子として P74-P77
 ・シリアル・アレイ・ユニット 0 (SAU0) のチャンネル 2 の CS110
 ・送信/受信を続けて行うため、DMA コントローラの DMA0, DMA1

ウェイト用タイマ
 ・5ms 以上のプログラム内ウェイト用としてタイマ・アレイ・ユニット (TAU) のチャンネル 1 のインターバル・タイマ・モード

録音/再生キー
 ・キー割り込み入力端子 KR0, KR1

動作中インジケータ (LED) 出力
 ・動作中インジケータ用 LED 出力として P72 で出力

<使用する周辺の初期設定の主な設定内容>

割り込みの禁止

CPU/周辺ハードウェア・クロック周波数を X1 発振クロック (20MHz 使用) に設定

ポートの設定

オーディオ CODEC への設定

- ・オーディオ CODEC へのクロック供給
- ・プログラム内ウェイト用のタイマ設定
- ・オーディオ CODEC へのリセット信号出力
- ・レジスタ設定用の I2C インタフェース設定
- ・オーディオ CODEC 内のレジスタ設定にてシステム ON

オーディオ CODEC のオーディオ・データ用 I2S バス・インタフェース設定

- ・T000 出力 (16kHz) を使用し、LRCLK の出力設定
- ・CS100 の設定

EEPROM の SPI インタフェース設定

- ・データ送信用に CS110 の設定
- ・連続送信用として受信用に DMA0, 送信用に DMA1 を設定
- ・EEPROM の全消去 (オール OFFH)

キーの取り込み開始

割り込みの許可

<メイン処理の主な内容>

- ・ キー処理
- ・ 録音開始設定
- ・ 再生開始設定
- ・ 録音終了設定
- ・ 再生終了設定

<INTTM00 割り込み処理 (INTTM00 使用, LRCLK と BCLK, SDOUT および SDIN 同期用)の主な内容>

- ・ I2S バス・インタフェース動作開始

<INTCS100 割り込み処理 (INTCS100 使用, CSI 送信/受信用)の主な内容>

- ・ I2S バス・インタフェースでのデータ送受信
- ・ 受信データの保存
- ・ EEPROM へのライト開始
- ・ EEPROM へのリード開始

<INTCS110 割り込み処理 (INTCS110 使用, CSI 送信 / 受信用)の主な内容>

- ・ EEPROM への命令バイト送信
- ・ EEPROM への 24 ビット・アドレス送信
- ・ DMA0, DMA1 を使用した EEPROM からのデータ受信開始
- ・ DMA1 を使用した EEPROM へのデータ送信開始

<INTDMA0 割り込み処理 (INTDMA0 使用, CSI 受信用)の主な内容>

- ・ EEPROM からのリード開始
- ・ I2S バス・インタフェース動作開始
- ・ EEPROM の選択
- ・ I2S バス・インタフェース動作停止

<INTDMA1 割り込み処理 (INTDMA1 使用, CSI 送信用)の主な内容>

- ・ EEPROM の選択
- ・ I2S バス・インタフェース動作停止

***** /

/*=====

前処理指令 (#pragma 指令)

=====*/

```
#pragma SFR                /* 特殊機能レジスタ(SFR)名を記述可能にする */
#pragma DI                 /* DI 命令を記述可能にする */
#pragma EI                 /* EI 命令を記述可能にする */
#pragma NOP                /* NOP 命令を記述可能にする */
#pragma interrupt INTTM00  fn_inttm00  RB1 /* 割り込み関数宣言: INTTM00 */
#pragma interrupt INTCS100 fn_intcsi00 RB1 /* 割り込み関数宣言: INTCS100 */
#pragma interrupt INTCS110 fn_intcsi10 RB2 /* 割り込み関数宣言: INTCS110 */
#pragma interrupt INTDMA0  fn_intdma0  RB1 /* 割り込み関数宣言: INTDMA0 */
#pragma interrupt INTDMA1  fn_intdma1  RB1 /* 割り込み関数宣言: INTDMA1 */
```

/*=====

関数プロトタイプ宣言

=====*/

```
void fn_I2cWrite(unsigned char *addr, unsigned char size);
```

/*=====

ポート定義

=====*/

```
/* オーディオ CODEC リセット */
#define P_RESETB      P7.3      /* リセット */
#define PM_RESETB     PM7.3     /* リセット出力 */
```

```

/* オーディオ CODEC 内のレジスタ設定用 I2C インタフェース */
#define P_SCL          P6.0          /* SCL0 ラッチ */
#define PM_SCL         PM6.0         /* SCL0 出力 */
#define P_SDA          P6.1          /* SDA0 ラッチ */
#define PM_SDA        PM6.1         /* SDA0 入出力 */

/* オーディオ CODEC I2S バス・インタフェース */
#define P_LRCLK        P0.1          /* LRCLK */
#define PM_LRCLK       PM0.1         /* LRCLK 出力 */
#define P_SCK00        P1.0          /* SCK00 */
#define PM_SCK00       PM1.0         /* SCK00 出力 */
#define PM_S100        PM1.1         /* S100 入力 */
#define P_S000         P1.2          /* S000 */
#define PM_S000        PM1.2         /* S000 出力 */

/* キー入力 (2本使用) */
#define P_KEY          P7            /* KR(P70-P71 を使用) */
#define PM_KEY         PM7           /* KR 入力(PM70-PM71 を使用) */
#define P_PLAYKY       P7.0          /* 再生キー */
#define PM_PLAYKY      PM7.0         /* 再生キー入力 */
#define P_RECKY        P7.1          /* 録音キー */
#define PM_RECKY       PM7.1         /* 録音キー入力 */

/* 動作中インジケータ用 LED 出力 (1本使用) */
#define P_LED          P7.2          /* 動作中インジケータ(Low アクティブ) */
#define PM_LED         PM7.2         /* 動作中インジケータ(LED)出力 */
/* 動作中に点灯させ、キー入力が可能なときに消灯させます。 */

/* EEPROM SPI インタフェース (4個使用) */
#define P_CS           P7            /* チップ・セレクト CS(P73-P77 を使用) */
#define PM_CS          PM7           /* チップ・セレクト CS 出力(PM73-PM77 を使用) */
#define P_CS3          P7.7          /* CS3 */
#define PM_CS3         PM7.7         /* CS3 出力 */
#define P_CS2          P7.6          /* CS2 */
#define PM_CS2         PM7.6         /* CS2 出力 */
#define P_CS1          P7.5          /* CS1 */
#define PM_CS1         PM7.5         /* CS1 出力 */
#define P_CS0          P7.4          /* CS0 */
#define PM_CS0         PM7.4         /* CS0 出力 */
#define P_SCK10        P0.4          /* SCK10 */
#define PM_SCK10       PM0.4         /* SCK10 出力 */
#define PM_S110        PM0.3         /* S110 入力 */
#define P_S010         P0.2          /* S010 */
#define PM_S010        PM0.2         /* S010 出力 */

/* =====
RAM の定義
===== */
/* 全体の動作 */
unsigned char ucPlayMode;          /* 動作状態 */
#define CRESET          0          /* リセット */
#define CSTOP           1          /* 停止中 */
#define CREC            2          /* 録音中 */
#define CREC_END        3          /* 録音終了中 */
#define CPLAY_START     4          /* 再生準備開始 */
#define CPLAY_SET       5          /* 再生準備中 */
#define CPLAY           6          /* 再生中 */
#define CPLAY_END       7          /* 再生終了中 */

/* オーディオ CODEC との I2S バス・インタフェース動作関連 */
unsigned char *ucl2sAddress;        /* I2S 送受信データの保存先アドレス */
unsigned char ucl2sByteCounter;     /* I2S データの 1 バイト送受信カウンタ */
boolean bl2sMemoryPage;            /* I2S バスの送受信中のページ: ページ 1(0) / ページ 2(1) */

/* EEPROM との SPI インタフェース動作関連 */
unsigned short ushEepromAddress;    /* EEPROM のリード/ライト・アドレス(ushEepromAddress*0x100) */
unsigned char ucEepromCs;           /* EEPROM のチップ・セレクト CS 設定値 */
unsigned char ucEepromSeq;          /* EEPROM の転送シーケンス */
#define CEEPSEQ_RESET   0          /* リセット状態 */
#define CEEPSEQ_WREN    1          /* ライトイネーブル送信 */
#define CEEPSEQ_INST    2          /* 命令バイト送信

```

```
#define CEEPSEQ_ADDRH      3      /* 24 ビットアドレス上位 8 ビット部分送信 */
#define CEEPSEQ_ADDRM      4      /* 24 ビットアドレス中位 8 ビット部分送信 */
#define CEEPSEQ_ADDRLL     5      /* 24 ビットアドレス下位 8 ビット部分送信 */
#define CEEPSEQ_DATA       6      /* DMA によるデータ送信 / 受信 */
```

```
/* 内蔵 RAM のメモリ領域 */
unsigned char ucMemoryPage1[3*2*42]; /* ページ 1 (LRCLK の 42 クロック分) */
unsigned char ucMemoryPage2[3*2*42]; /* ページ 2 (LRCLK の 42 クロック分) */
```

```
/*=====
ROM の定義
=====*/
```

OKI セミコンダクタ製のオーディオ CODEC ML2612 の状態を遷移させるためのオーディオ CODEC 内のレジスタ設定を行うための設定値 ROM テーブル群となります。
I2C バス・インタフェース・ライト処理で使します。

【例】

```
{ 0xE5, 0x0000111B, 0, 0 } ;Trimming 1
,{ 0xE9, 0x0000001B, 0, 0 } ;Trimming 2
-----|-----|~|
| | | +----- ウェイト情報(注)
| | | +----- ライト・データ
+----- ライト・アドレス
```

注) データをライトした後のウェイト時間を示します。
5ms × (記述した数値) の時間でウェイトを行います。
0 の場合はウェイトなしとなります。

このサンプル・プログラムの設定データは、録音・再生を行うためのほんの一例です。詳細はデータ・シート等を参考にしてください。

```
-----*/
static const unsigned char aSystemOnTbl[][3]=
{
/*-----
パワーOFF システム ON
-----*/
/*-----
Trimming
-----*/
{ 0x0E5, 0b00000111, 0 } /* Trimming 1 */
,{ 0x0E9, 0b00000001, 0 } /* Trimming 2 */
/*-----
Digital Block ON
-----*/
,{ 0x001, 0x003, 0 } /* Sampling Rate */
,{ 0x003, 0x001, 0 } /* PLLNL */
,{ 0x005, 0x000, 0 } /* PLLNH */
,{ 0x007, 0x000, 0 } /* PLLML */
,{ 0x009, 0x002, 0 } /* PLLMH */
,{ 0x00B, 0x002, 0 } /* PLLDIV */
,{ 0x00F, 0b00000001, 0 } /* CLK Input/Output control */
,{ 0x00D, 0b00000001, 0 } /* Clock Enable */
,{ 0x00D, 0b00000101, 150/5 } /* Clock Enable */
};
```

```
static const unsigned char aRecordOnTbl[][3]=
{
/*-----
システム ON 録音開始
-----*/
/*-----
Record Function Setting
-----*/
/* ボリューム設定 */
{ 0x033, 0x03F, 0 } /* Mic Input Volume */
,{ 0x039, 0b00000000, 0 } /* Mic Boost Volume */
```

```

,{      0x049 ,0b00000001 ,0 } /* AMP Volume Control Function Enable */
,{      0x04B ,0b00000000 ,0 } /* Amplifier Volume Fader Control */
,{      0x069 ,0b00011010 ,0 } /* Volume Control Func Enable */
,{      0x06B ,0b00000010 ,0 } /* Mixer & Volume Control */
,{      0x06D ,0x0FF ,0 } /* Playback Digital Volume Control */

/* HPF1 ON */
,{      0x067 ,0b00000001 ,0 } /* Filter Func Enable */

/* HPF2 ON */
// ,{      0x067 ,0b00000000 ,0 } /* Filter Func Enable */
// ,{      0x07F ,0b00000000 ,0 } /* HPF2 CutOff */

/* Programmable Equalizer 設定 */
// ,{      0x06D ,0b11111111 ,0 } /* Record Digital Volume */
// ,{      0x067 ,0b00000001 ,0 } /* Filter Func Enable */
// ,{      0x067 ,0b00000000 ,0 } /* Filter Func Enable */
// ,{      0x07F ,0b00000000 ,0 } /* HPF2 CutOff */
// ,{      0x067 ,0b00000000 ,0 } /* Filter Func Enable */
// ,{      0x075 ,0b00000000 ,0 } /* EQ gain1 Band0 */
// ,{      0x077 ,0b00000000 ,0 } /* EQ gain1 Band1 */
// ,{      0x079 ,0b00000000 ,0 } /* EQ gain1 Band2 */
// ,{      0x07B ,0b00000000 ,0 } /* EQ gain1 Band3 */
// ,{      0x07D ,0b00000000 ,0 } /* EQ gain1 Band4 */
// ,{      0x083 ,0b00000000 ,0 } /* EQ Band0 Coef0H */
// ,{      0x085 ,0b00000000 ,0 } /* EQ Band0 Coef1L */
// ,{      0x087 ,0b00000000 ,0 } /* EQ Band0 Coef1H */
// ,{      0x089 ,0b00000000 ,0 } /* EQ Band1 Coef0L */
// ,{      0x08B ,0b00000000 ,0 } /* EQ Band1 Coef0H */
// ,{      0x08D ,0b00000000 ,0 } /* EQ Band1 Coef1L */
// ,{      0x08F ,0b00000000 ,0 } /* EQ Band1 Coef1H */
// ,{      0x091 ,0b00000000 ,0 } /* EQ Band2 Coef0L */
// ,{      0x093 ,0b00000000 ,0 } /* EQ Band2 Coef0H */
// ,{      0x095 ,0b00000000 ,0 } /* EQ Band2 Coef1L */
// ,{      0x097 ,0b00000000 ,0 } /* EQ Band2 Coef1H */
// ,{      0x099 ,0b00000000 ,0 } /* EQ Band3 Coef0L */
// ,{      0x09B ,0b00000000 ,0 } /* EQ Band3 Coef0H */
// ,{      0x09D ,0b00000000 ,0 } /* EQ Band3 Coef1L */
// ,{      0x09F ,0b00000000 ,0 } /* EQ Band3 Coef1H */
// ,{      0x0A1 ,0b00000000 ,0 } /* EQ Band4 Coef0L */
// ,{      0x0A3 ,0b00000000 ,0 } /* EQ Band4 Coef0H */
// ,{      0x0A5 ,0b00000000 ,0 } /* EQ Band4 Coef1L */
// ,{      0x0A7 ,0b00000000 ,0 } /* EQ Band4 Coef1H */

/* ALC 設定 */
,{      0x069 ,0b00000010 ,0 } /* Volume Control Func Enable */
// ,{      0x0B1 ,0b00000000 ,0 } /* ALC Mode */
// ,{      0x0B3 ,0b00000000 ,0 } /* ALC Attack Time */
// ,{      0x0B5 ,0b00000000 ,0 } /* ALC Decay Time */
// ,{      0x0B7 ,0b00000000 ,0 } /* ALC Hold Time */
// ,{      0x0B9 ,0b00000000 ,0 } /* ALC Target Level */
// ,{      0x0BB ,0b00000000 ,0 } /* ALC Max/Min Gain */
// ,{      0x0BD ,0b00000000 ,0 } /* Noise Gate Threshold */
// ,{      0x0BF ,0b00000000 ,0 } /* ALC Zero Cross Time OutPlayback Limiter Control Register */
// ,{      0x02F ,0b00000000 ,0 } /* ZC-CMP Power Management

/*-----
Record Analog Block ON
-----*/

/* VMID 生成回路 ON */
,{      0x021 ,0b00000001 ,5/5 } /* Reference Power Management */
,{      0x021 ,0b00000010 ,5/5 } /* Reference Power Management */

/* マイクバイパス 回路 ON */
,{      0x021 ,0b00000110 ,0 } /* Reference Power Management */

/* MIN 入力設定 (差動入力) */
,{      0x05B ,0b00000010 ,0 } /* Mic IF Control */

/* Power Management Register */
,{      0x023 ,0b00001000 ,0 } /* Input Power Management */
,{      0x023 ,0b000001010 ,0 } /* Input Power Management */

```

```

/*-----
Record start
-----*/
,{ 0x069 ,0b00010000 ,0 } /* Volume Control Func Enable */
,{ 0x069 ,0b00010000 ,0 } /* Volume Control Func Enable */
,{ 0x013 ,0b00000001 ,5/5 } /* Record/Playback Run */
,{ 0x06B ,0b00110010 ,0 } /* Mixer & Volume Control */
,{ 0x069 ,0b00011000 ,0 } /* Volume Control Func Enable */
,{ 0x069 ,0b00001000 ,0 } /* Volume Control Func Enable */
};

static const unsigned char aRecordOffTbl[][3]=
{
/*-----
録音 システム ON
-----*/
/*-----
Record stop
-----*/
{ 0x06B ,0b00000010 ,0 } /* Mixer & Volume Control */
,{ 0x069 ,0b00001000 ,0 } /* Volume Control Func Enable */
,{ 0x069 ,0b00011000 ,200/5 } /* Volume Control Func Enable */
,{ 0x013 ,0b00000000 ,10/5 } /* Record/Playback Run */

/*-----
Record Analog Block OFF
-----*/
,{ 0x023 ,0b00001000 ,0 } /* Input Power Management */
,{ 0x023 ,0b00000000 ,0 } /* Input Power Management */
,{ 0x021 ,0b00000001 ,0 } /* Reference Power Management */
,{ 0x021 ,0b00000000 ,0 } /* Reference Power Management */
};

static const unsigned char aPlayOnTbl[][3]=
{
/*-----
システム ON 再生開始
-----*/
/*-----
Playback Function Setting
-----*/
/* ボリューム設定 */
{ 0x03B ,0b00000000 ,0 } /* Speaker AMP Volume */
,{ 0x0C9 ,0b00000000 ,0 } /* Boost Volume */
,{ 0x049 ,0b00000000 ,0 } /* AMP Volume Control Function Enable */
,{ 0x04B ,0b00000000 ,0 } /* Amplifier Volume Fader Control */
,{ 0x069 ,0b00000000 ,0 } /* Volume Control */
,{ 0x06B ,0b00000010 ,0 } /* Mixer & Volume */
,{ 0x071 ,0b11111111 ,0 } /* Playback Digital Volume Control */

/* Programmable Equalizer 設定 */
// ,{ 0x067 ,0b00000000 ,0 } /* Filter Func Enable */
// ,{ 0x075 ,0b00000000 ,0 } /* EQ gain1 Band0 */
// ,{ 0x077 ,0b00000000 ,0 } /* EQ gain1 Band1 */
// ,{ 0x079 ,0b00000000 ,0 } /* EQ gain1 Band2 */
// ,{ 0x07b ,0b00000000 ,0 } /* EQ gain1 Band3 */
// ,{ 0x07d ,0b00000000 ,0 } /* EQ gain1 Band4 */
// ,{ 0x083 ,0b00000000 ,0 } /* EQ Band0 Coef0H */
// ,{ 0x085 ,0b00000000 ,0 } /* EQ Band0 Coef1L */
// ,{ 0x087 ,0b00000000 ,0 } /* EQ Band0 Coef1H */
// ,{ 0x089 ,0b00000000 ,0 } /* EQ Band1 Coef0L */
// ,{ 0x08B ,0b00000000 ,0 } /* EQ Band1 Coef0H */
// ,{ 0x08D ,0b00000000 ,0 } /* EQ Band1 Coef1L */
// ,{ 0x08F ,0b00000000 ,0 } /* EQ Band1 Coef1H */
// ,{ 0x091 ,0b00000000 ,0 } /* EQ Band2 Coef0L */
// ,{ 0x093 ,0b00000000 ,0 } /* EQ Band2 Coef0H */
// ,{ 0x095 ,0b00000000 ,0 } /* EQ Band2 Coef1L */
// ,{ 0x097 ,0b00000000 ,0 } /* EQ Band2 Coef1H */
// ,{ 0x099 ,0b00000000 ,0 } /* EQ Band3 Coef0L */
// ,{ 0x09B ,0b00000000 ,0 } /* EQ Band3 Coef0H */
// ,{ 0x09D ,0b00000000 ,0 } /* EQ Band3 Coef1L */
// ,{ 0x09F ,0b00000000 ,0 } /* EQ Band3 Coef1H */
// ,{ 0x0A1 ,0b00000000 ,0 } /* EQ Band4 Coef0L */

```



```

// ,{ 0x0A3 ,0b00000000 ,0 } /* EQ Band4 Coef0H */
// ,{ 0x0A5 ,0b00000000 ,0 } /* EQ Band4 Coef1L */
// ,{ 0x0A7 ,0b00000000 ,0 } /* EQ Band4 Coef1H */

/* Playback Limiter 設定 */
,{ 0x069 ,0b00011001 ,0 } /* Volume Control Func Enable */
// ,{ 0x0C1 ,0b00000000 ,0 } /* Attack Time */
// ,{ 0x0C3 ,0b00000000 ,0 } /* Decay Time */
// ,{ 0x0C5 ,0b00000000 ,0 } /* Target Level */
// ,{ 0x0C7 ,0b00000000 ,0 } /* Max/Min Gain */

/*-----
Playback Analog Block ON
-----*/
/* VMID 生成回路 ON */
,{ 0x021 ,0b00000001 ,5/5 } /* Reference Power Management */
,{ 0x021 ,0b00000010 ,5/5 } /* Reference Power Management */

/* ミュート設定 */
,{ 0x049 ,0b00000000 ,0 } /* AMP Volume Control Function Enable */
,{ 0x049 ,0b00000010 ,0 } /* AMP Volume Control Function Enable */

/* DAC 設定 */
,{ 0x025 ,0b00000010 ,0 } /* DAC Power */
,{ 0x055 ,0b00000010 ,0 } /* Speaker AMP Output Control */

/* スピーカ・アンプ ON */
,{ 0x027 ,0x013 ,0 } /* Power Management */
,{ 0x027 ,0x01F ,0 } /* Reference Power Management */
,{ 0x03B ,0x033 ,0 } /* Speaker AMP Volume */
,{ 0x04B ,0b00000000 ,0 } /* Amplifier Volume Fader Control */
,{ 0x049 ,0b00000011 ,0 } /* AMP Volume Control Function Enable */
,{ 0x049 ,0b00000001 ,500/5 } /* AMP Volume Control Function Enable */

/*-----
Playback start
-----*/
,{ 0x069 ,0b00011101 ,0 } /* Volume Control Func Enable */
,{ 0x069 ,0b00010001 ,0 } /* Volume Control Func Enable */
,{ 0x013 ,0b00000010 ,5/5 } /* Record/Playback Run */
,{ 0x06B ,0b00110010 ,0 } /* Mixer & Volume Control */
,{ 0x069 ,0b00011001 ,0 } /* Volume Control Func Enable */
,{ 0x069 ,0b00001001 ,0 } /* Volume Control Func Enable */
};

static const unsigned char aPlayOffTbl[][3]=
{
/*-----
再生 システム ON
-----*/
/*-----
Playback Stop
-----*/
{ 0x06B ,0b00000010 ,0 } /* Mixer & Volume Control */
,{ 0x069 ,0b00001000 ,0 } /* Volume Control Func Enable */
,{ 0x069 ,0b00011000 ,100/5 } /* Volume Control Func Enable */
,{ 0x013 ,0b00000000 ,0 } /* Record/Playback Run */

/*-----
Playback Analog Block OFF
-----*/
/* スピーカアンプ BTL モード OFF */
,{ 0x04B ,0b00000000 ,0 } /* Amplifier Volume Fader Control */
,{ 0x049 ,0b00000001 ,0 } /* AMP Volume Control Function Enable */
,{ 0x049 ,0b00000011 ,100/5 } /* AMP Volume Control Function Enable */
,{ 0x027 ,0x013 ,0 } /* AMP Power Management */
,{ 0x027 ,0x000 ,0 } /* AMP Power Management */
,{ 0x025 ,0b00000000 ,0 } /* DAC Power Management */
,{ 0x021 ,0b00000000 ,0 } /* Reference Power Management */
};

```

```

/*****
使用する周辺の初期設定

*****/
void hdwinit(void)
{
/*-----
割り込み禁止
-----*/
DI();
/*-----
クロック周波数の設定
-----
20MHz の X1 発振回路で動作が行えるように設定します
-----*/
CMC = 0b01000001;          /* クロック動作モード */
/*      ||| ||| | +----- AMPH: 10MHz < fMX 20MHz */
/*      ||| | +----- <000> */
/*      ||| +----- OSCSELS: P123/P124 端子を入力ポート */
/*      || +----- <0> */
/*      ++----- EXCLK/OSCSEL: X1 発振モード(20MHz) */

CSC = 0b01000000;          /* クロック動作ステータス制御 */
/*      ||| ||| | +----- HIOSTOP: 高速内蔵発振回路動作 */
/*      || +----- <00000> */
/*      | +----- XTSTOP: XT1 発振回路停止 */
/*      +----- MSTOP: X1 発振回路動作 */

OSMC = 0b00000001;          /* 動作スピード・モード */
/*      ||| ||| | +----- FSEL: 10MHz を越える周波数で動作 */
/*      +----- <00000> */

OSTS = 0b00000101;          /* 発振安定時間: 2^15/fX */

while(!OSTC.2) {}          /* クロック発振安定待ち */

CKC = 0b00011000;          /* クロック選択 */
/*      ||| ||| +----- MDIV2-0: CPU/周辺ハードウェア・クロック(fCLK)=fMX */
/*      ||| +----- <1> */
/*      ||| +----- MCMO: 高速システム・クロック(fMX) */
/*      || +----- <R> */
/*      | +----- CSS: メイン・システム・クロック(fMAIN)=fCLK */
/*      +----- <R> */

/*-----
ポート 0 の設定
-----*/
P0 = 0b00011010;          /* P00,P02,P05-P06 の出力ラッチ Low, P01,P03-P04 の出力ラッチ HIGH */
PM0 = 0b10000000;          /* P00-P06 を出力ポートに設定 */
/* P01: オーディオ CODEC への LRCLK 出力 */
/* P02: EEPROM への S010 出力 */
/* P03: EEPROM への S110 入力 */
/* P04: EEPROM への SCK10 出力 */
/* P00,P05-P06: 未使用 */

/*-----
ポート 1 の設定
-----*/
P1 = 0b00000000;          /* P10-P17 の出力ラッチ Low */
PM1 = 0b00000000;          /* P10-P17 を出力ポートに設定 */
/* P10: オーディオ CODEC への SCK00 出力 */
/* P11: オーディオ CODEC への S100 入力 */
/* P12: オーディオ CODEC への S000 出力 */
/* P13-P17: 未使用 */

/*-----
ポート 2 の設定
-----*/
P2 = 0b00000000;          /* P20-P27 の出力ラッチ Low */

```

```

PM2 = 0b00000000;          /* P20-P27 を出力ポートに設定 */
                             /* P20-P27: 未使用 */

/*-----*/
ポート 3 の設定
-----*/
P3 = 0b00000000;          /* P30-P31 の出力ラッチ Low */
PM3 = 0b11111100;          /* P30-P31 を出力ポートに設定 */
                             /* P30-P31: 未使用 */

/*-----*/
ポート 4 の設定
-----*/
P4 = 0b00000000;          /* P40-P47 の出力ラッチ Low */
PM4 = 0b00000000;          /* P40-P47 を出力ポートに設定 */
                             /* P40-P47: 未使用 */

/*-----*/
ポート 5 の設定
-----*/
P5 = 0b00000000;          /* P50-P57 の出力ラッチ Low */
PM5 = 0b00000000;          /* P50-P57 を出力ポートに設定 */
                             /* P50-P57: 未使用 */

/*-----*/
ポート 6 の設定
-----*/
P6 = 0b00000000;          /* P60-P67 の出力ラッチ Low */
PM6 = 0b00000000;          /* P60-P67 を出力ポートに設定 */
                             /* P60: オーディオ CODEC 内のレジスタ設定のための SCL0 出力 */
                             /* P61: オーディオ CODEC 内のレジスタ設定のための SDA0 入出力 */
                             /* P62-P67: 未使用 */

/*-----*/
ポート 7 の設定
-----*/
P7 = 0b00000000;          /* P70-P77 の出力ラッチ Low */
PM7 = 0b00000011;          /* P70-P71 を入力ポート, P72-P77 を出力ポートに設定 */
                             /* P70: 再生キー入力 */
                             /* P71: 録音キー入力 */
                             /* P72: 動作中インジケータ(LED)出力 */
                             /* P73: オーディオ CODEC へのリセット出力 */
                             /* P74: EEPROM への CS0 出力 */
                             /* P75: EEPROM への CS1 出力 */
                             /* P76: EEPROM への CS2 出力 */
                             /* P77: EEPROM への CS3 出力 */

/*-----*/
ポート 8 の設定
-----*/
P8 = 0b00000000;          /* P80-P87 の出力ラッチ Low */
PM8 = 0b00000000;          /* P80-P87 を出力ポートに設定 */
                             /* P80-P87: 未使用 */

/*-----*/
ポート 11 の設定
-----*/
P11 = 0b00000000;          /* P110-P111 の出力ラッチ Low */
PM11 = 0b11111100;          /* P110-P111 を出力ポートに設定 */
                             /* P110-P111: 未使用 */

/*-----*/
ポート 12 の設定
-----*/
P12 = 0b00000000;          /* P120 の出力ラッチ Low */
PM12 = 0b11111110;          /* P120 を出力ポートに設定 */
                             /* P120: 未使用 */

/*-----*/
ポート 13 の設定
-----*/
P13 = 0b00000000;          /* P130-P131 の出力ラッチ Low */
PM13 = 0b11111100;          /* P131-P131 を出力ポートに設定 */

```

/* P130-P131: 未使用 */

/*-----
 ポート 14 の設定
 -----*/

P14 = 0b00000000; /* P140-P145 の出力ラッチ Low */
 PM14 = 0b11000000; /* P140-P145 を出力ポートに設定 */
 /* P140-P145: 未使用 */

/*-----
 ポート 15 の設定
 -----*/

P15 = 0b00000000; /* P150-P157 の出力ラッチ Low */
 PM15 = 0b00000000; /* P150-P157 を出力ポートに設定 */
 /* P150-P157: 未使用 */

/*-----
 オーディオ CODEC 内のレジスタ設定
 -----*/

下記動作を行います。
 ・オーディオ CODEC へのクロック供給
 ・プログラム内ウエイト用のタイマ設定
 ・オーディオ CODEC へのリセット信号出力
 ・オーディオ CODEC 内のレジスタ設定用の I2C インタフェース設定

/* タイマ・アレイユニットのタイマ・クロック選択 */
 TAUQEN = 1; /* タイマ・アレイユニットの入力クロック供給 */
 TPSOL = 0b10010000; /* 動作クロック選択 */
 /* |||++++----- CK00: fCLK */
 /* +----- CK01: fCLK/2^9 */

/*-----
 オーディオ CODEC へのクロック供給
 -----*/

TO02 出力(10MHz)
 -----*/
 TMR02 = 0b0000000000000000; /* 動作モード設定 */
 /* |||++++----- MD023-020: インターバル・タイマ・モード */
 /* |||++++----- <00 固定> */
 /* |||++++----- CIS021-020: 未使用 */
 /* |||++++----- STS022-020: ソフトウェア・トリガ・スタートのみ有効 */
 /* |||++++----- MASTER02: 単体動作 */
 /* |||++++----- CSS02: CKS02 ビットで指定したマクロ・クロック MCK */
 /* |||++++----- <00 固定> */
 /* |||++++----- CKS02: PRS レジスタで設定した動作クロック CK00 */

TDR02 = 0; /* インターバル設定 : 10MHz 出力(20MHz) */
 TOEOL.2 = 1; /* TO02 出力 */
 TSOL.2 = 1; /* 動作開始(トリガ・ビット) */

/*-----
 プログラム内で使用するウエイト用
 -----*/

TM01 使用
 -----*/
 TMR01 = 0b1000000000000000; /*動作モード設定 */
 /* |||++++----- MD013-010: インターバル・タイマ・モード */
 /* |||++++----- <00 固定> */
 /* |||++++----- CIS011-010: 未使用 */
 /* |||++++----- STS012-010: ソフトウェア・トリガ・スタートのみ有効 */
 /* |||++++----- MASTER01: 単体動作 */
 /* |||++++----- CSS01: CKS01 ビットで指定したマクロ・クロック MCK */
 /* |||++++----- <00 固定> */
 /* |||++++----- CKS01: PRS レジスタで設定した動作クロック CK01 */

/* 電源安定待ち */
 TDR01 = 195*210/5-1; /* インターバル設定(210ms) */
 TSOL.1 = 1; /* タイマ動作開始 */
 TMIF01 = 0; /* INTTM00 割り込み要求クリア */
 while(!TMIF01) {}

```

/* リセット */
P_RESETB = 0;
/* ウエイト約 5us 以上*/
TDR01 = 0; /* インターバル設定 */
TSOL.1 = 1; /* タイマ動作開始 */
TMIF01 = 0; /* INTTM00 割り込み要求クリア */
while(!TMIF01) {}
P_RESETB = 1;

/* 電源安定待ち */
TDR01 = 195*160/5-1; /* インターバル設定(160ms) */
TSOL.1 = 1; /* タイマ動作開始 */
TMIF01 = 0; /* INTTM00 割り込み要求クリア */
while(TMIF01) {}
TTOL.1 = 1; /* タイマ動作停止 */

/*-----
オーディオ CODEC 内のレジスタ設定値の送信
-----*/
IIC0 使用
-----*/
IICOEN = 1; /* シリアル・インタフェース IIC0 の入力クロック供給 */

IICE0 = 0; /* 動作停止 */

P_SCL = 0; /* SCL0 ラッチ */
PM_SCL = 0; /* SCL0 端子 */
P_SDA = 0; /* SDA0 端子 */
PM_SDA = 0; /* SDA0 端子 */

IICX0 = 0b00000000; /* 転送クロック選択 */
/* |||||+----- CLX0 */
/* ++++++----- <0000000> */
IICCL0 = 0b00001110;
/* |||||++----- CL01-00(+CLX0): fCLK/96(ファースト・モード) */
/* ||||+----- DFC0: デジタル・フィルタ・オン */
/* |||+----- SMC0: ファースト・モードで動作 */
/* ||+----- DADO: <R>SDAO 端子のレベル検出 */
/* ||+----- CLD0: <R>SCL0 端子のレベル検出 */
/* ++----- <00> */

IICF0 = 0b00000011; /* 通信予約禁止 */
/* |||||+----- IICRSV: 通信予約機能禁止 */
/* |||||+----- STCEN: 初期スタート許可 */
/* ||++++----- <0000 固定> */
/* |+----- IICBSY: <R>IIC バス状態フラグ */
/* +----- STCF: <R>STT0 クリアフラグ */

IICCO = 0b00001000; /* マスタ時の初期設定 */
/* |||||+----- SPT0: ストップ・コンディション・トリガ */
/* |||||+----- STT0: スタート・コンディション・トリガ */
/* ||||+----- ACKE0: アクノリッジ制御 */
/* |||+----- WTIMO: ウエイトおよび割込要求発生制御:9クロック */
/* ||+----- SPIE0: ストップコンディション検出による割込要求発生禁止 */
/* |+----- WRELO: ウエイト解除しない */
/* |+----- LRELO: 通信退避:通常動作 */
/* +----- IICE0: I2C の動作許可 */

SPT0 = 1; /* ストップ・コンディション設定 */

/* オーディオ CODEC 内のレジスタ設定: システム ON 設定 */
fn_I2cWrite(&aSystemOnTbl[0][0], sizeof(aSystemOnTbl));

/*-----
オーディオ CODEC のオーディオ・データ用 I2S バス・インタフェース設定
-----*/

下記動作を行います。
・ LRCLK 出力用に T000 出力と INTTM00 割り込みの設定
・ データ送受信用に CS100 の設定
-----*/

```

```

/*-----
LRCLK の出力設定
-----*/
T000 出力(16kHz)
-----*/
TMR00 = 0b0000000000000000; /* 動作モード設定 */
/*      |||+----- MD003-000: インターバル・タイマ・モード */
/*      |||+----- <00 固定> */
/*      |||+----- CIS001-000: 未使用 */
/*      |||+----- STS002-000: ソフトウェア・トリガ・スタートのみ有効 */
/*      |||+----- MASTER00: 単体動作 */
/*      ||+----- CSS00: CKS00 ビットで指定したマクロ・クロック MCK */
/*      ||+----- <00 固定> */
/*      ++----- CKS00: PRS レジスタで設定した動作クロック CK00 */
TDR00 = 625-1; /* インターバル設定: 16kHz 出力(32kHz) */

TMPR100 = 0; /* 優先順位を最高位に設定 */
TMPR000 = 0;

/*-----
CS100 の設定
-----*/
SAU0EN = 1; /* シリアル・アレイ・ユニットの入力クロック供給 */

NOP(); /* ウェイト */
NOP();
NOP();
NOP();

SPS0L = 0b00000000; /* 動作クロック選択 fCLK */
/*      |||+----- PRS003-00: fCLK */
/*      +++----- PRS013-10: 未使用 */

SDR00 = (12-1) << 9; /* ビット 15-7: 転送クロック設定(833kHz) */

SMR00 = 0b0000000001000001; /* 動作モード選択:CSI モード */
/*      |||+----- MD000: バッファ空き割り込み */
/*      |||+----- MD002-1: CSI モード */
/*      |||+----- <100 固定> */
/*      |||+----- SIS000: 未使用 */
/*      |||+----- <0 固定> */
/*      |||+----- STS00: ソフトウェア・トリガのみ有効(CSI 時は固定) */
/*      ||+----- <00000 固定> */
/*      |+----- CSS00: 転送クロックを CKS00 ビットで指定した動作クロック MCK の分周クロックに設定 */
/*      |----- CKS00: 動作クロックを PRS レジスタで設定したプリスケアラ出力クロック CK00 に設定 */
/*      +-----

/* 初期データ出力 (CS110 の設定時に同時に行う) */
/* S00 = 0b0000000100000001; /* S0 端子&SCK 端子の初期出力設定 */
/*      |||+----- S00n-0: 送信で使用 */
/*      +----- CK0n-0: チャンネル n-0 のシリアル・クロック出力 */

SOE0L.0 = 1; /* 出力許可 */

P_SCK00 = 1; /* SCK00 ラッチ:ハイ */
PM_SCK00 = 0; /* SCK00 端子出力設定 */
PM_S100 = 1; /* S100 端子入力設定 */
P_S000 = 1; /* S000 ラッチ:ハイ */
PM_S000 = 0; /* S000 端子出力設定 */

SS0L.0 = 1; /* CS100 動作開始(トリガ・ビット) */
CS1IF00 = 0; /* INTCS100 割り込み要求クリア */
CS1MK00 = 0; /* INTCS100 割り込み処理許可 */

CS1PR100 = 0; /* 優先順位を INTT00 の次に設定 */
CS1PR000 = 1;

```

/*-----
EEPROM 用の SPI インタフェース設定
-----*/

下記動作を行います。


```
/* 初期データ出力 */
S00 = 0b0000000100000001; /* S0 端子&SCK 端子の初期出力設定 */
/* ||| ||| |||+++++++----- S0n-0: 送信で使用 */
/* ++++++----- CK0n-0: チャンネル n-0 のシリアル・クロック出力 */
S0EOL.2 = 1; /* 出力許可 */

P_SCK10 = 1; /* SCK00 ラッチ:ハイ */
PM_SCK10 = 0; /* SCK00 端子出力設定 */
PM_S110 = 1; /* S100 端子入力設定 */
P_S010 = 1; /* S000 ラッチ:ハイ */
PM_S010 = 0; /* S000 端子出力設定 */

SS0L.2 = 1; /* CSI10 動作開始(トリガ・ビット) */

/*-----
EEPROM の全消去(オール OFFH)
-----*/
/* EEPROM ライトイネーブル */
P_CS &= 0b00001111; /* 全 EEPROM の選択 */
CSIIF10 = 0;
SIO10 = 0b00000110; /* 送信データ・セット(ライトイネーブル(WREN)命令, INTCS110 発生) */
while(!CSIIF10) {}
P_CS |= 0b11110000; /* 設定完了 */

/* EEPROM 全消去 */
P_CS &= 0b00001111; /* 全 EEPROM の選択 */
CSIIF10 = 0;
SIO10 = 0b11000111; /* 送信データ・セット(CE 全消去, INTCS110 発生) */
while(!CSIIF10) {}
P_CS |= 0b11110000; /* 設定完了 */

ST0L.2 = 1; /* CSI10 動作停止(トリガ・ビット) */
CS1MK10 = 1; /* INTCS110 割り込み処理禁止 */

/*-----
キーの取り込み開始
-----*/

KRM = 0b00000011; /* KRO-KR1 有効 */
KRIF = 0; /* INTKR 割り込み要求クリア */
KRMK = 1; /* INTKR 割り込み処理禁止 */

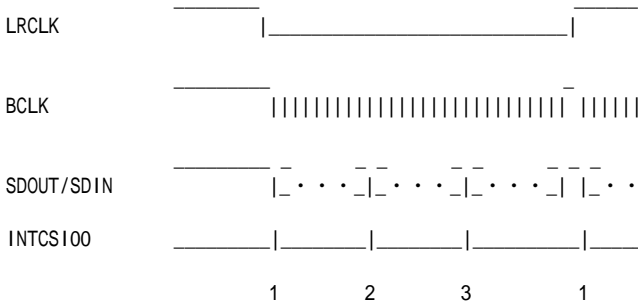
P_LED = 1; /* 動作中インジケータ OFF(キー入力可能) */

/*-----
割り込み許可
-----*/
EI();
}
```


/*-----*/

INTCS100 割り込み処理
(INTCS100 使用, CSI 送信/受信)

オーディオ CODEC との I2S バス・インタフェースで、データの送受信を行います。録音中、CSI00 を使用し、受信データを保存します。通常はダミー・データをセットすることでデータの連続受信を行います。LRCLK との同期を行うため、3 バイト毎に受信のみを行い、連続受信の動作を停止させます。停止させた動作は、INTTMO0 割り込み処理で動作再開となり LRCLK と同期させることができます。受信したデータが 1 ページに達した場合は、EEPROM へのライト動作を開始します。再生中も録音中と同様に、CSI00 を使用し、データを連続送信しますが、LRCLK との同期を行うため、3 バイト毎に連続送信の動作を停止させます。停止させた動作は、INTTMO0 割り込み処理で動作再開となり LRCLK と同期させることができます。送信したデータが 1 ページに達した場合は、EEPROM から別の 1 ページ分のデータのリードを開始します。



レジスタバンク 1 使用

/*-----*/

```
__interrupt void fn_intcsi00(void)
{
    ucI2sByteCounter++;          /* CSI00 送受信カウンタ更新 */

    if(ucPlayMode==CREC){
        /*-----
        録音中
        -----*/
        /* ページの最終位置の確認 */
        if(bI2sMemoryPage){
            /* ページ 2 使用中 */
            if(ucI2sAddress>=&ucMemoryPage2[3*2*42]){
                /* ページ 2 の最終 */
                ucI2sAddress = &ucMemoryPage1[0];          /* I2S 受信データの保存先アドレス: ページ 1 */
                bI2sMemoryPage = 0;

                /* EEPROM へのライト開始 */
                ucEepromSeq = CEEPSEQ_WREN;                /* ライトイネーブル(WREN)送信 */
                P_CS = (P_CS & 0b00001111)|ucEepromCs; /* EEPROM CS 設定 */
                CS1IF10 = 0;                                /* INTCS110 割り込み要求クリア */
                CS1MK10 = 0;                                /* INTCS110 割り込み処理許可 */
                SSOL.2 = 1;                                  /* CS110 動作開始(トリガ・ビット) */
                SI010 = 0b00000110;                         /* 送信データ・セット(ライトイネーブル(WREN)命令, INTCS110 発生) */
            }
        }
        else{
            /* ページ 1 使用中 */
            if(ucI2sAddress>=&ucMemoryPage1[3*2*42]){
                /* ページ 1 の最終 */
                ucI2sAddress = &ucMemoryPage2[0];          /* I2S 受信データの保存先アドレス: ページ 2 */
                bI2sMemoryPage = 1;

                /* EEPROM へのライト開始 */
                ucEepromSeq = CEEPSEQ_WREN;                /* ライトイネーブル(WREN)送信 */
                P_CS = (P_CS & 0b00001111)|ucEepromCs; /* EEPROM CS 設定 */
                CS1IF10 = 0;                                /* INTCS110 割り込み要求クリア */
                CS1MK10 = 0;                                /* INTCS110 割り込み処理許可 */
                SSOL.2 = 1;                                  /* CS110 動作開始(トリガ・ビット) */
            }
        }
    }
}
```



```

/*****

```

```

INTCSI10 割り込み処理
(INTCSI10 使用, CSI 送信 / 受信)

```

```

-----
録音中は, EEPROM への命令バイトの送信と 24 ビット・アドレスの送信を行い,
EEPROM へのデータ送信のタイミングで, DMA のチャンネル 1 を開始し, CSI10 の
シングル送信にて続けて送信します。

```

```

再生中は, EEPROM への 24 ビット・アドレスの送信を行い, EEPROM からのデー
タ受信のタイミングで, DMA のチャンネル 0 とチャンネル 1 を開始し, CSI10 のシン
グル受信にて続けて受信します。ただし, DMA のチャンネル 1 はダミー・データ
のセット用となります。

```

```

レジスタバンク 1 使用

```

```

*****/

```

```

__interrupt void fn_intcsi10(void)
{
    EI();          /* 多重割り込みを許可 */

    if(ucPlayMode==CREC){
/*-----
録音中
-----*/
        ucEepromSeq++;          /* EEPROM の転送シーケンス更新 */
        /* 命令バイト送信 */
        switch(ucEepromSeq){
            case CEEPSEQ_INST:
                /* 命令バイト送信タイミング */
                P_CS |= 0b11110000;          /* ノン・アクティブ */
                NOP();
                P_CS = (P_CS & 0b00001111)|ucEepromCs;          /* EEPROM 選択 */
                SIO10 = 0b00000010;          /* ライト指定送信 */
                break;

            case CEEPSEQ_ADDRH:
                /* 24 ビット・アドレス上位送信タイミング */
                SIO10 = (unsigned char)(ushEepromAddress>>8);
                break;

            case CEEPSEQ_ADDRM:
                /* 24 ビット・アドレス中位送信タイミング */
                SIO10 = (unsigned char)(ushEepromAddress);
                break;

            case CEEPSEQ_ADDRL:
                /* 24 ビット・アドレス下位送信タイミング */
                SIO10 = 0x000;
                break;

            case CEEPSEQ_DATA:
            default:
                /* データ送信開始 */
                /* DMA 転送開始 */
                if(bl2sMemoryPage){
                    /* ページ 2 使用中 */
                    DRA1 = (unsigned short)&ucMemoryPage1[1];
                    /* 送信バッファをページ 1 に設定 */
                }
                else{
                    /* ページ 1 使用中 */
                    DRA1 = (unsigned short)&ucMemoryPage2[1];
                    /* 送信バッファをページ 2 に設定 */
                }
                DBC1 = 3*2*42-1;          /* 1 ページ分 */
                DST1 = 1;          /* DMA1 転送許可(CSI10 送信用) */

                DMAIF1 = 0;          /* INTDM1 割り込み要求クリア */
                DMAMK1 = 0;          /* INTDM1 割り込み処理許可 */
                CSIIF10 = 0;          /* INTCSI10 割り込み要求クリア */
                CSIMK10 = 1;          /* INTCSI10 割り込み処理禁止 */
        }
    }
}

```



```

DBC0 = 3*2*42;          /* ページ・サイズ設定 */

/* DMA 転送開始 */
DST1 = 1; /* DMA1 転送許可(CS110 送信用) */
DST0 = 1; /* DMA0 転送許可(CS110 受信用) */

DMAMK1 = 1; /* INTDM1 割り込み処理禁止 */
CS1IF10 = 0; /* INTCS110 割り込み要求クリア */
CS1MK10 = 1; /* INTCS110 割り込み処理禁止 */

DMAIF0 = 0; /* INTDM0 割り込み要求クリア */
DMAMK0 = 0; /* INTDM0 割り込み処理許可 */

SSOL.2 = 1; /* CS110 動作開始(トリガ・ビット) */
SI010 = 0x0FF; /* ダミー・データのセット(INTCS100 発生) */
break;
}
}
}

/*****

INTDMA0 割り込み処理
(INTDMA0 使用, CS1 受信用)

-----

再生中, EEPROM から 1 ページ分のデータを受信完了したときに発生する割り込み
処理です。使用する 4 個の EEPROM の選択, I2S バス・インタフェースの動作開始と
動作停止を行います。

レジスタバンク 1 使用

*****/
__interrupt void fn_intdma0(void)
{
/* 通信動作の終了待ちの完了待ち */
while(SSR02 & 0b0000000010000000) {}

/* 1 ページ分のデータ受信完了 */
P_CS |= 0b11110000; /* ノン・アクティブ */
ucEepromSeq = CEEPSEQ_INST;

if(ucPlayMode==CPLAY){
/* 再生中 */
/*-----
I2S バス・インタフェース動作開始
-----*/
/* EEPROM の選択 */
if((ucEepromCs & 0b10000000)==0b00000000){
/* EEPROM が一周 */
ushEepromAddress++;
if(ushEepromAddress>=128000/256){
/* I2S バス・インタフェース動作停止 */
P_LRCLK = 1; /* LRCLK 出力無効化(ハイ) */
TTOL.0 = 1; /* T000 出力動作停止(トリガ・ビット) */
STOL.0 = 1; /* CS1000 出力動作停止(トリガ・ビット) */
STOL.2 = 1; /* CS1010 出力動作停止(トリガ・ビット) */
ucPlayMode = CPLAY_END; /* 再生終了中 */
}
else{
ucEepromCs = 0b11100000;
}
}
else{
ucEepromCs = (ucEepromCs|0b00001000) <<1;
}
}
else if(ucPlayMode==CPLAY_START){
/* 再生準備開始 */
/* EEPROM からのリード開始 */
ucPlayMode = CPLAY_SET; /* 再生準備 */
P_CS = (P_CS & 0b00001111)|0b11010000; /* EEPROM 選択 */
}
}
}

```

```

    b12sMemoryPage = 0;
    CS1IF10 = 0;          /* INTCSI10 割り込み要求クリア */
    CS1MK10 = 0;         /* INTCSI10 割り込み処理許可 */
    SSOL.2 = 1;         /* CS110 動作開始(トリガ・ビット) */
    SI010 = 0b00000011; /* 送信データ・セット(リード指定, INTCSI10 発生) */
}
else if(ucPlayMode==CPLAY_SET){
/* 再生準備中 */
    ucPlayMode = CPLAY;          /* 再生中 */
    ucI2sAddress = &ucMemoryPage1[1]; /* 送信バッファの先頭を設定 */
    b12sMemoryPage = 0;         /* ページ1送信 */
    /* I2S バス・インタフェース動作開始 */
    SSOL.0 = 1;                 /* CS100 動作開始(トリガ・ビット) */
    P_LRCLK = 0;                /* LRCLK の動作開始 */
    TSOL.0 = 1;                 /* T000 出力動作開始(トリガ・ビット) */
    SI000 = ucMemoryPage1[0];   /* 送信データ・セット(INTCSI00 発生) */
}
DST1 = 0;          /* DMA1 転送停止 */
DST0 = 0;          /* DMA0 転送停止 */
DMAMKO = 1;       /* INTDMO 割り込み処理禁止 */
}

```

```

/*****

```

INTDMA1 割り込み処理
(INTDMA1 使用, CSI 送信用)

録音中, EEPROM へ 1 ページ分のデータを送信完了したときに発生する割り込み処理です。使用する 4 個の EEPROM の選択と I2S バス・インタフェースの動作停止を行います。

レジスタバンク 1 使用

```

*****/

```

```

__interrupt void fn_intdma1(void)
{
    /* 通信動作の終了待ちの完了待ち */
    while(SSR02 & 0b000000001000000) {}

    /* 1 ページ分のデータ送信完了 */
    P_CS |= 0b11110000; /* 1 ページ分のデータ送信完了 */
    ucEepromSeq = CEEPSEQ_RESET;

    /* EEPROM の選択 */
    if((ucEepromCs & 0b10000000)==0b00000000){
        ushEepromAddress++;
        if(ushEepromAddress>=128000/256){
            /* I2S バス・インタフェース動作停止 */
            P_LRCLK = 1; /* LRCLK 出力無効化(ハイ) */
            TTOL.0 = 1; /* T000 出力動作停止(トリガ・ビット) */
            STOL.0 = 1; /* CS1000 出力動作停止(トリガ・ビット) */
            STOL.2 = 1; /* CS1010 出力動作停止(トリガ・ビット) */
            ucPlayMode = CPLAY_END; /* 再生停止 */
        }
        else{
            ucEepromCs = 0b11100000;
        }
    }
    else{
        ucEepromCs = (ucEepromCs|0b00001000) <<1;
    }
    DST1 = 0; /* DMA1 転送停止 */
    DST0 = 0; /* DMA0 転送停止 */
    DMAMK1 = 1; /* INTDM1 割り込み処理禁止 */
}

```

```

/*****

```

オーディオ CODEC 内のレジスタ設定用 I2C バス・インタフェース・ライト処理

I2C バス・インタフェース IIC0 の機能を使用し、オーディオ CODEC 内のレジスタの設定データの書き込みを行います。

```

-----
[ I N ] *addr  : レジスタ設定値 ROM テーブルの先頭アドレス上位 4 ビット
        size  : ROM テーブルのサイズ
[ OUT ] -

*****/
void fn_I2cWrite(unsigned char *addr, unsigned char size)
{
    register unsigned char cnt;

    IICE0 = 1;          /* 動作許可 */

    for(cnt=0; cnt<=size/3; cnt++){
        /* スタート・コンディション発行 */
        STTO = 1;

        /* スレーブ・アドレス送信 */
        while(!STDO) {} /* アドレス送信期間? No, */
        IICIF0 = 0;
        IIC0 = 0b00110100; /* アドレス送信開始(ライト指定) */
        while(!IICIF0) {} /* 送信完了? No, */
        if(ACKD0){
            /* ライト・アドレス送信 */
            IICIF0 = 0; /* INTIIC10 割り込み要求クリア */
            IIC0 = *addr; /* データ送信開始 */
            while(!IICIF0) {} /* 送信完了? No, */

            if(ACKD0){
                /* ライト・データ送信 */
                addr++; /* 参照位置更新 */
                IICIF0 = 0; /* INTIIC10 割り込み要求クリア */
                IIC0 = *addr; /* データ送信開始 */
                while(!IICIF0) {} /* 送信完了? No, */

                if(ACKD0){
                    /* ストップ・コンディション発行 */
                    SPT0 = 1;

                    /* ウェイト制御 */
                    addr++; /* 参照位置更新 */
                    if(*addr!=0){
                        TDR01 = 195*(*addr)-1; /* インターバル設定(5ms*addr) */
                        TMIF01 = 0; /* INTTM01 割り込み要求クリア */
                        TSOL.1 = 1; /* タイマ動作開始 */
                        while(!TMIF01) {} /* インターバル経過? No, */
                        TTOL.1 = 1; /* タイマ動作停止 */
                    }
                    addr++; /* 参照位置更新 */
                }
            }
            else{
                SPT0 = 1; /* ストップ・コンディション発行 */
                cnt--; /* 再送信 */
                addr--;
            }
        }
        else{
            SPT0 = 1; /* ストップ・コンディション発行 */
            cnt--; /* 再送信 */
        }
    }
}
}

```

付録B 改版履歴

版 数	発行年月	改版箇所	改版内容
第1版	November 2008	-	-

【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

—— お問い合わせ先 ——

【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.necel.co.jp/>

【営業関係，技術関係お問い合わせ先】

半導体ホットライン

（電話：午前 9:00～12:00，午後 1:00～5:00）

電 話 : 044-435-9494

E-mail : info@necel.com

【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか，NECエレクトロニクスの販売特約店へお申し付けください。
