

お客様各位

カタログ等資料中の旧社名の扱いについて

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

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

2010年4月1日
ルネサスエレクトロニクス株式会社

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

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

ご注意書き

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

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

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

アプリケーション・ノート

78K0R/LH3

サンプル・プログラム（温度測定）

A/Dコンバータの機能を使用した温度測定編

この資料は、温度測定を行いその温度に合わせてLEDの輝度制御を行うサンプル・プログラムの動作概要や使用方法を説明したものです。温度測定では12ビット分解能のA/Dコンバータにて測定し、測定結果を摂氏温度に変換してRAMに保持、その値に対応してLEDの輝度制御をPWM出力にて行っています。

対象デバイス

μ PD78F1506 , 78F1507 , 78F1508

目次

- 第1章 概要 ... 3
- 第2章 回路イメージ ... 5
 - 2.1 回路イメージ ... 5
 - 2.2 マイコン以外の使用デバイス ... 6
 - 2.3 端子機能一覧 ... 6
- 第3章 ソフトウェアについて ... 7
 - 3.1 ファイル構成 ... 7
 - 3.2 使用する内蔵周辺機能 ... 8
 - 3.3 初期設定と動作概要 ... 9
 - 3.4 フロー・チャート ... 10
- 第4章 設定方法について ... 12
 - 4.1 周辺ハードウェア・マクロ使用の設定 ... 12
 - 4.2 タイマ・アレイ・ユニット0の設定 ... 13
 - 4.3 ボルテージ・リファレンスの設定 ... 25
 - 4.4 A/Dコンバータの設定 ... 27
 - 4.5 ソフトウェア記述例 ... 33
 - 4.6 温度測定処理 ... 37
 - 4.7 温度計算式 ... 43
 - 4.8 乗除算器の使用法 ... 45
 - 4.9 PWM機能としての動作 ... 47
 - 4.10 PWM出力によるLEDの輝度制御 ... 50
- 第5章 関連資料 ... 51
- 付録A プログラム・リスト ... 52
- 付録B 改版履歴 ... 116

資料番号 U2003JJ1V0AN00 (第1版)

発行年月 September 2009 N

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

(M8E0909J)

第1章 概 要

このサンプル・プログラムは、温度測定、およびLEDの輝度制御を行えるようにするための使用例を示しています。

サンプル・プログラムでは、A/Dコンバータ、およびボルテージ・リファレンスの機能を使用した温度測定と、タイマ・アレイ・ユニットのPWM機能を使用したLEDの輝度の制御を行います。温度測定の結果は、摂氏温度としてRAMに保存します。測定できる温度の範囲は - 34.08 から 99.95 です。LEDの輝度は、温度測定の結果を使用したPWM出力で制御します。

(1) 初期設定の主な内容

< オプション・バイトでの設定 >

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

< リセット解除後の初期化処理での設定 >

入出力ポートの設定

- ・ P33をLEDの輝度制御用にタイマ出力端子に設定
- 低電圧検出回路^{注1}の機能を使用し、2.7V以上の電源電圧を確保
- CPUクロックをX1発振回路動作に設定 (20MHz)

高速内蔵発振回路の停止

タイマ・アレイ・ユニット0の設定^{注2}

- ・ チャンネル0を温度測定のタイミング用に周期が約500msのインターバル・タイマ・モードに設定
- ・ チャンネル6, 7をLEDの輝度制御用にPWM機能に設定

ボルテージ・リファレンスの設定

- ・ 基準電圧源としてボルテージ・リファレンスを選択
- ・ ボルテージ・リファレンスの出力電圧を2.0Vに設定
- ・ タイマ・アレイ・ユニット0のチャンネル1を使用し、ボルテージ・リファレンスの動作安定待ちとして約17msをウエイト

A/Dコンバータの設定

- ・ P21/ANI1端子をアナログ入力チャンネル用に設定
- ・ 動作モードをソフトウェア・トリガ・モード (シングル変換モード) に設定
- ・ A/D変換時間を $240/f_{CLK}$ (約12 μ s) に設定
- ・ - 側の基準電圧源をAV_{SS}, + 側の基準電圧源をボルテージ・リファレンス出力 (V_{REFOUT}) に設定

注1. 低電圧検出回路についての詳細は、ユーザズ・マニュアルを参照してください。

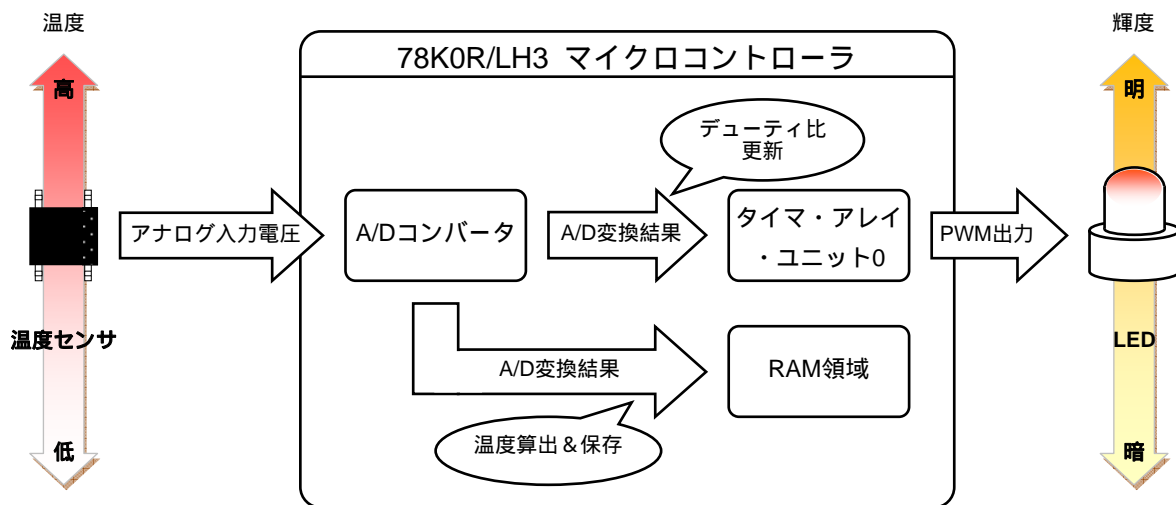
2. タイマ・アレイ・ユニット0をインターバル・タイマとして使用するための設定についての詳細は、ユーザズ・マニュアルを参照してください。

(2) メイン・ループ以降の内容

初期設定完了後，約500ms周期でA/Dコンバータの機能を使用した温度測定，およびタイマ・アレイ・ユニット0のチャンネル0の機能を使用したLEDの輝度制御を行います。温度測定の結果は，摂氏温度としてRAMに保存します。測定できる温度の範囲は - 34.08 から99.95 です。LEDの輝度は，温度測定の結果を使用したPWM出力で制御します。

温度などの計算には，乗除算器を使用しています。

【動作概略図】

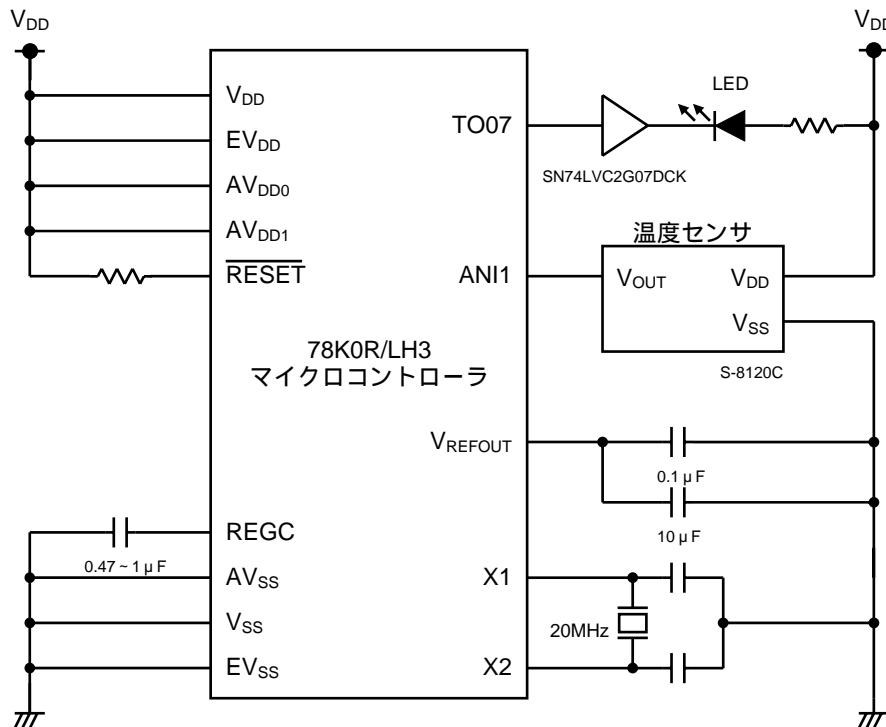


第2章 回路イメージ

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

2.1 回路イメージ

回路イメージを次に示します。



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

2. EV_{DD} , AV_{DD0} , および AV_{DD1} は, V_{DD} と同電位にしてください。
3. AV_{SS} は EV_{SS} , V_{SS} と同電位にし, GND に直接接続してください。
4. ボルテージ・リファレンス動作時は, 基準電圧安定化用に, V_{REFOUT}/AV_{REFP} 端子に必ずタンタル・コンデンサ (容量: $10 \mu F \pm 30\%$, ESR: 2 (MAX.), ESL: 10nH (MAX.)) とセラミック・コンデンサ (容量: $0.1 \mu F \pm 30\%$, ESR: 2 (MAX.), ESL: 10nH (MAX.)) を接続してください。また, ボルテージ・リファレンス動作時に, V_{REFOUT}/AV_{REFP} 端子から電圧を印加しないでください。
5. REGCはコンデンサ ($0.47 \sim 1 \mu F$) を介し, V_{SS} に接続してください。
6. 回路イメージ中に記載のない未使用端子は以下のように処理してください。
出力ポート : 出力モードに設定し, オープン (未接続) にしてください
入力ポート : 個別に抵抗を介して, V_{DD} または V_{SS} に接続してください
7. このサンプル・プログラムでは, P40/TOOL0端子, およびP41/TOOL1端子をオンチップ・デバッグ用に使用します。

2.2 マイコン以外の使用デバイス

マイコン以外の使用デバイスを次に示します。

(1) LED

タイマ・アレイ・ユニットのPWM機能で輝度を制御します。

(2) オープンドレイン・バッファIC 74LVC07

PWM出力で制御するLEDのバッファとして使用します。

(3) 温度センサ S-8120C

A/Dコンバータでの温度測定用に使用します。セイコーインスツル株式会社製の温度センサです。

2.3 端子機能一覧

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	
ANI1	A/Dコンバータのアナログ入力	P21/AMP00
TO07	タイマ出力 (PWM)	P33/INTP3/TI07

第3章 ソフトウェアについて

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

3.1 ファイル構成

ダウンロードする圧縮ファイルのファイル構成は，次のようになっています。

ファイル名	説明	同封圧縮 (*.zip) ファイル	
			
main.asm (アセンブリ言語版) ----- main.c (C言語版)	マイコンのハードウェア初期化処理と，メイン処理のソース・ファイル	注	注
op.asm	オプション・バイト設定用アセンブラ・ソース・ファイル (ウォッチドッグ・タイマの設定，高速内蔵発振クロック周波数の選択，LVIデフォルト・スタート機能の設定，オンチップ・デバッグ動作の設定などを行います。)		
78K0R_Lx3_Sensor.prw	統合開発環境 PM+用ワーク・スペース・ファイル		
78K0R_Lx3_Sensor.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 :
インターバル・タイマに設定し、温度測定の周期約500msを生成します。
- ・タイマ・アレイ・ユニット0 (TAU0) のチャンネル1 :
インターバル・タイマに設定し、約17msのボルテージ・リファレンスの動作安定待ちに使用します。
- ・タイマ・アレイ・ユニット0 (TAU0) のチャンネル6, 7 :
PWM機能に設定し、タイマ出力をLEDの輝度の制御に使用します。
- ・ボルテージ・リファレンス :
A/Dコンバータの2.0Vの基準電圧を生成します。
- ・A/Dコンバータ :
温度センサの出力電圧を測定します。
- ・乗除算器 :
A/D変換の結果から摂氏温度を算出するための計算等に使用します。
- ・低電圧検出回路 :
2.7V V_{DD} の確認用に使用します。
- ・端子機能 :
使用する端子機能を以下の表に示します。

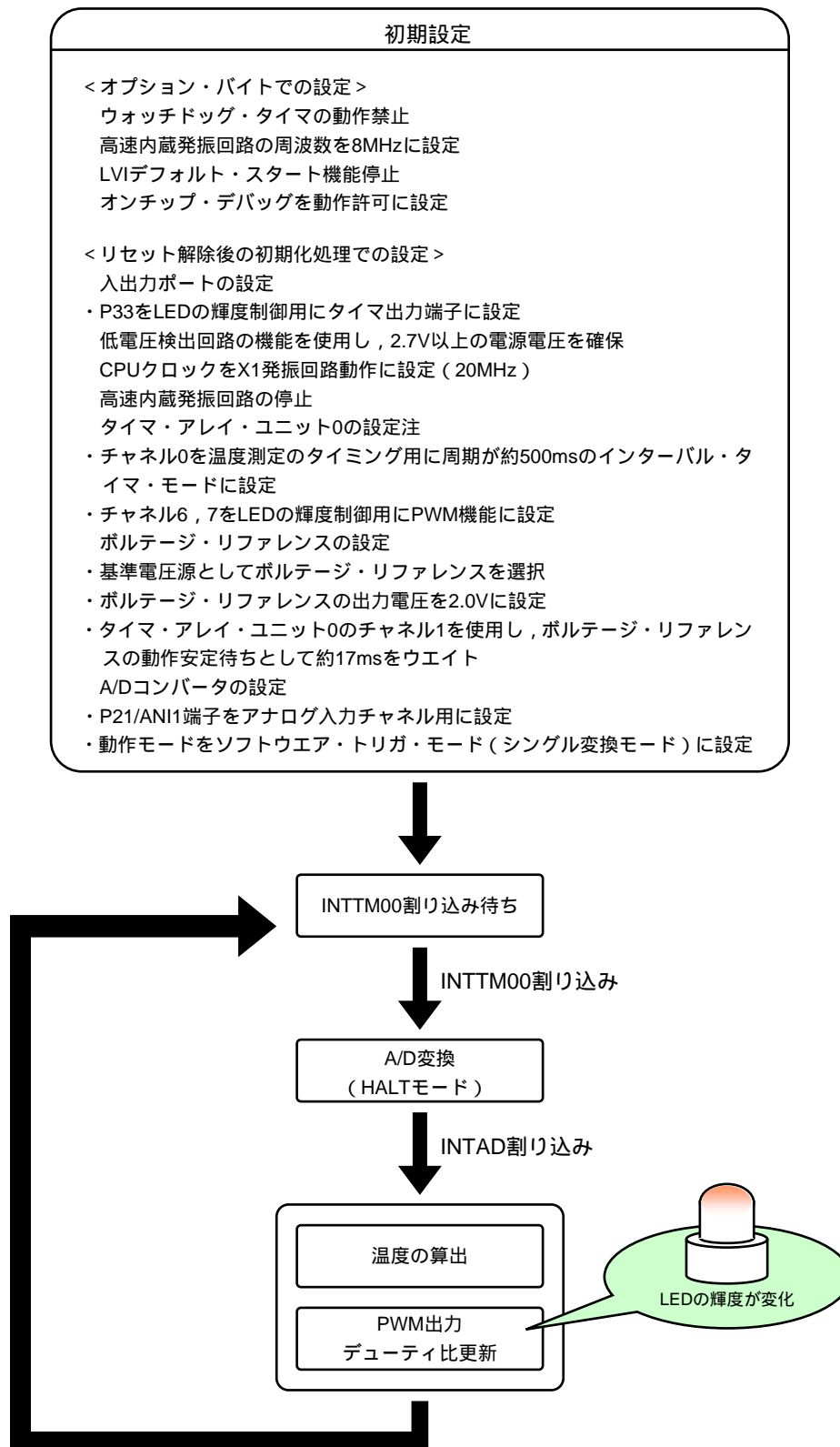
外部デバイス接続時の端子機能		兼用端子
名称	機能	
ANI1	A/Dコンバータのアナログ入力	P21/AMP00
TO07	タイマ出力 (PWM)	P33/INTP3/TI07

3.3 初期設定と動作概要

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

初期設定完了後は、タイマ・アレイ・ユニット0が生成する約500msの周期でA/D変換を行い、温度の算出とLEDの輝度制御を行います。温度測定結果はA/D変換の結果から乗除算器を使用して算出し、RAMに保存します。

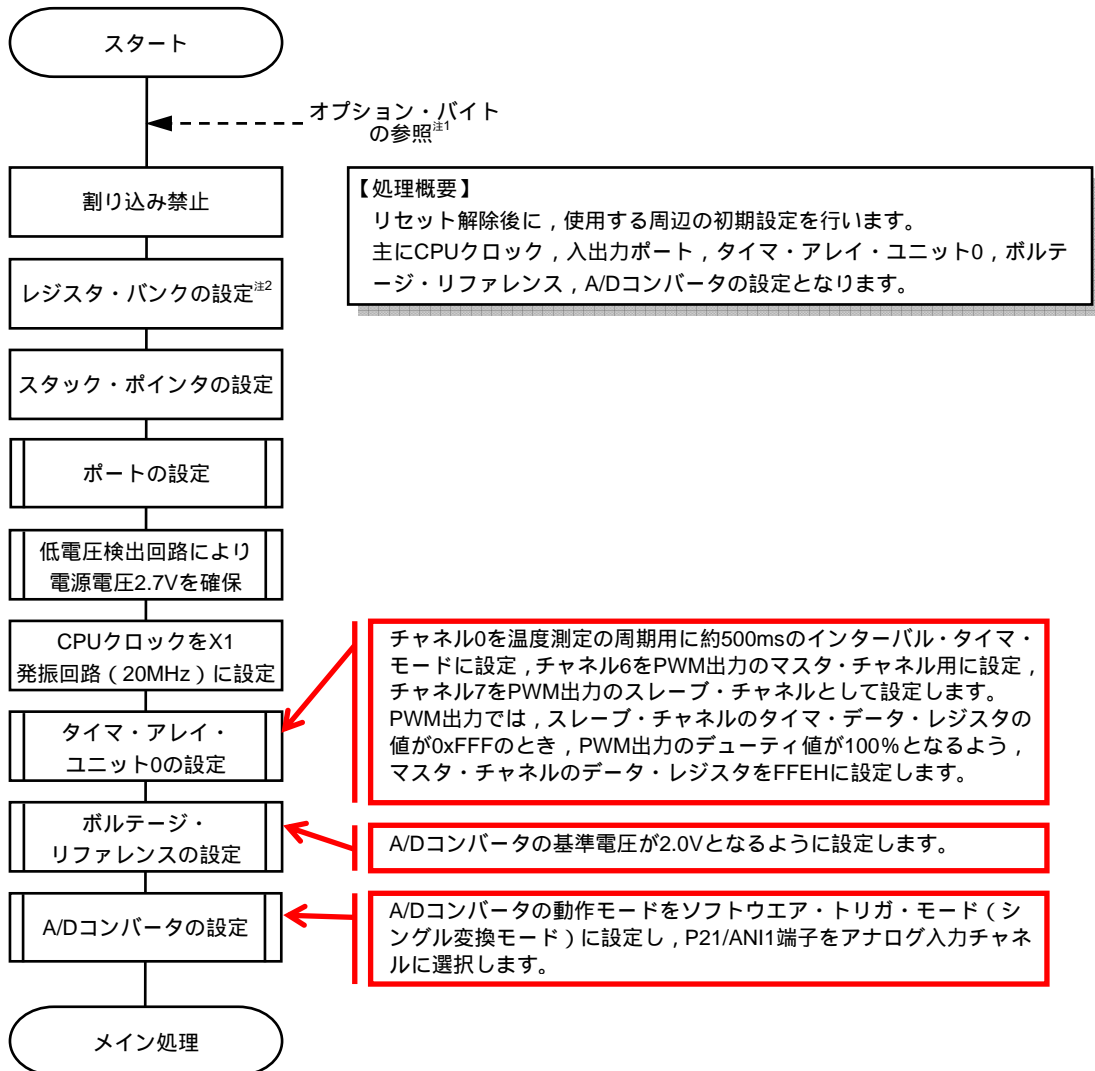
詳細については、次の状態遷移図（ステート・チャート）に示します。



3.4 フロー・チャート

このサンプル・プログラムのフロー・チャートを次に示します。

<リセット解除後の初期化処理での設定>



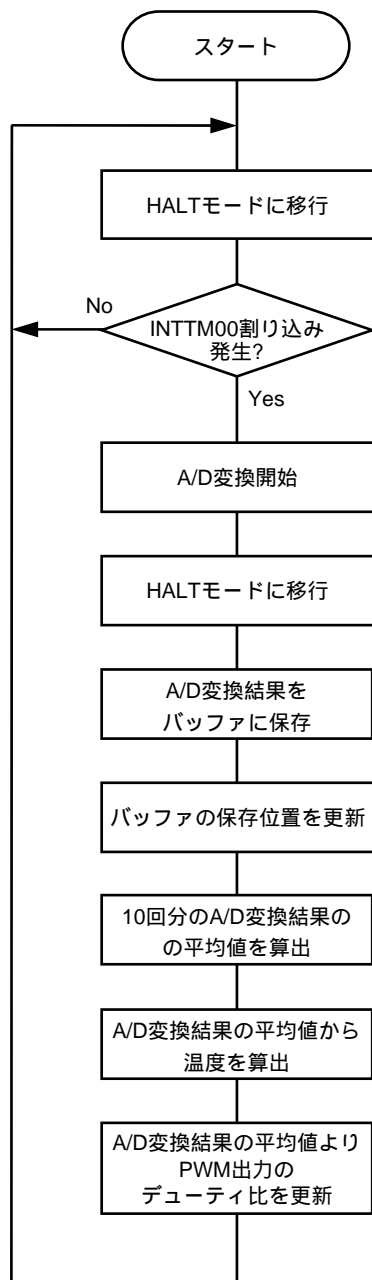
注1. オプション・バイトの参照は、リセット解除後にマイコンが自動的に行います。このサンプル・プログラムでは、オプション・バイトで以下の設定を行います。

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

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

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

<メイン処理>



【処理概要】

約500ms周期でA/D変換を行い、温度の算出とLEDの輝度制御を行います。

約500ms周期のINTTM00割り込み信号が発生するまでHALTモードに移行します。INTTM00割り込みが発生するとHALTモードを解除し、A/D変換、温度の算出、LEDの輝度制御を行います。

A/D変換の精度を向上させるため、A/D変換中はHALTモードに移行します。A/D変換終了割り込み要求信号が発生すると、HALTモードを解除します。

温度測定の精度を向上させるため、バッファに保存した10回のA/D変換結果から平均値を算出し、温度の算出とLEDの輝度制御を行います。A/D変換結果の平均値から乗除算器を使用して摂氏温度を算出し、RAMに保存します。

LED輝度制御では、PWM出力のデューティ比をA/D変換結果の平均値により更新します。

注意 C言語版のサンプル・プログラムで乗除算器を使用する場合、PM+のオプションで設定を行いますので、ソース・プログラム(main.c)には記述しません。PM+のコンパイラオプションの設定からスタートアップ・ルーチンのタブを選択し、「乗除算器を使用する」にチェックを入れることで、乗除算器を使用する実行ファイルが生成されます。

第4章 設定方法について

この章では、周辺ハードウェア・マクロ使用の設定、タイマ・アレイ・ユニット0の設定、ボルテージ・リファレンスの設定、A/Dコンバータの設定、ソフトウェア記述例、温度測定処理、温度計算式、乗除算器の使用方法、PWM機能としての動作、およびPWM出力によるLEDの輝度制御について説明します。

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

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

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

4.1 周辺ハードウェア・マクロ使用の設定

周辺ハードウェア・マクロ使用の設定は、次のレジスタを使用します。

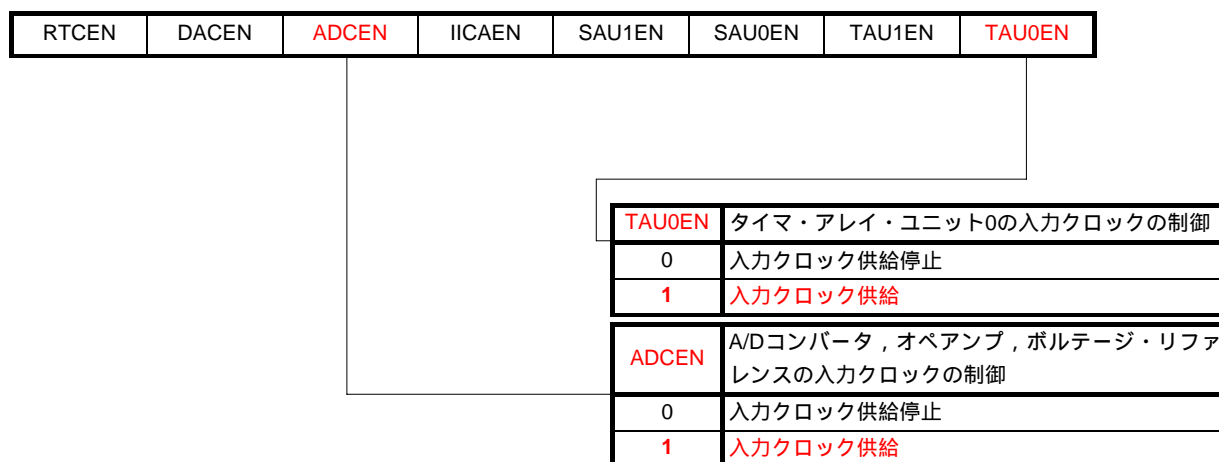
・周辺イネーブル・レジスタ0 (PER0)

(1) 周辺イネーブル・レジスタ0 (PER0)

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

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

図4-1 周辺イネーブル・レジスタ0 (PER0) のフォーマット



備考 図の赤字部分がサンプル・プログラムでの設定値となります。

4.2 タイマ・アレイ・ユニット0の設定

タイマ・アレイ・ユニット0はチャンネル0を温度測定のタイミング用にインターバル・タイマ・モードで使用し、チャンネル6, 7をPWM出力に使用します。

ここではタイマ・アレイ・ユニット0をPWM出力として使用する場合の設定方法を説明します。インターバル・タイマとして使用するための設定についての詳細は、ユーザズ・マニュアルを参照してください。

タイマ・アレイ・ユニット0のPWM出力では、次の11種類のレジスタを使用します。

- ・周辺イネーブル・レジスタ0 (PER0)
- ・タイマ・クロック選択レジスタ0 (TPS0)
- ・タイマ・モード・レジスタ06, 07 (TMR06, TMR07)
- ・タイマ・データ・レジスタ06, 07 (TDR06, TDR07)
- ・タイマ出力レジスタ0 (TO0)
- ・タイマ出力許可レジスタ0 (TOE0)
- ・タイマ出力レベル・レジスタ0 (TOL0)
- ・タイマ出力モード・レジスタ0 (TOM0)
- ・タイマ・チャンネル開始レジスタ0 (TS0)
- ・ポート・レジスタ (P3)
- ・ポート・モード・レジスタ (PM3)

【PWM出力を行う場合のタイマ・アレイ・ユニット0の設定手順例】

- 周辺イネーブル・レジスタ0 (PER0) のビット0 (TAU0EN) をセット (1) (4.1参照)
- ポート・モード・レジスタ3 (PM3) でP33端子を出力モードに設定
- タイマ・クロック選択レジスタ0 (TPS0) でCK01を f_{CLK} に設定
- タイマ・モード・レジスタ06 (TMR06) でチャンネル6の動作クロックをCK01, 動作モードをインターバル・タイマ・モードに設定
- タイマ・モード・レジスタ07 (TMR07) でチャンネル7の動作クロックをCK01, 動作モードをワンカウント・モードに設定
- タイマ・データ・レジスタ06 (TDR06) で、マスタ・チャンネルのパルス周期としてFFE_Hを設定
- タイマ・データ・レジスタ07 (TDR07) でデューティ値が100%となるように設定
- タイマ出力モード・レジスタ0 (TOM0) でチャンネル6のタイマ出力モードをトグル・モードに設定
- タイマ出力モード・レジスタ0 (TOM0) でチャンネル7のタイマ出力モードを連動動作モードに設定
- タイマ出力レベル・レジスタ0 (TOL0) でチャンネル6, 7のタイマ出力レベルを正論理出力 (アクティブ・ハイ) に設定
- タイマ出力レジスタ0 (TO0) でチャンネル6のタイマ出力値を“0”に設定
- タイマ出力レジスタ0 (TO0) でチャンネル7のタイマ出力値を“1”に設定
- タイマ出力許可レジスタ0 (TOE0) でカウント動作によるTO06出力動作停止を設定
- タイマ出力許可レジスタ0 (TOE0) でカウント動作によるTO07出力動作許可を設定
- ポート・レジスタ3 (P3) でP33の出力ラッチに0を設定
- タイマ・チャンネル開始レジスタ0 (TS0) でチャンネル6とチャンネル7を動作許可に設定

(1) タイマ・クロック選択レジスタ0 (TPS0)

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

タイマ動作中のTPS0の書き換えは、次の場合のみ可能です。

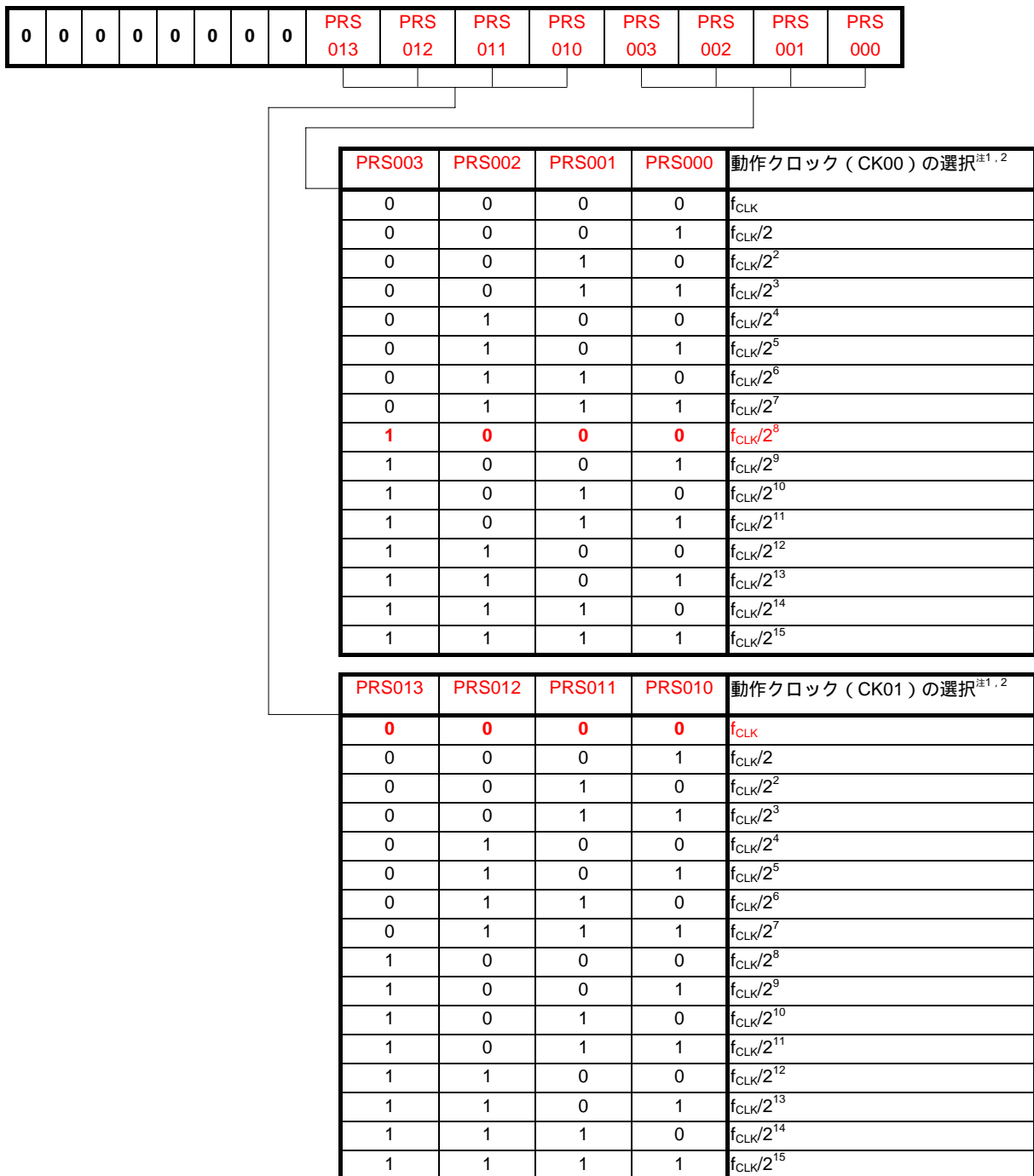
PRS000-PRS003ビットの書き換え：

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

PRS010-PRS013ビットの書き換え：

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

図4 - 2 タイマ・クロック選択レジスタ0 (TPS0) のフォーマット



(注の説明, 注意, 備考は次ページにあります。)

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

注意 ビット15-8には、必ず0を設定してください。

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

(2) タイマ・モード・レジスタ06, 07 (TMR06, TMR07)

TMR06, TMR07は、タイマ・アレイ・ユニット0のチャンネル6およびチャンネル7の動作モード設定レジスタです。動作クロック (MCK) の選択, カウント・クロックの選択, マスタ/スレーブの選択, スタート・トリガとキャプチャ・トリガの設定, タイマ入力の有効エッジ選択, 動作モード (インターバル, キャプチャ, イベント・カウンタ, ワンカウント, キャプチャ&ワンカウント) 設定を行います。

図4-3 タイマ・モード・レジスタ06 (TMR06) のフォーマット (1/2)

CKS06	0	0	CCS06	MAST ER06	STS062	STS061	STS060	CIS061	CIS060	0	0	MD063	MD062	MD061	MD060																																																																																																			
<table border="1" style="width:100%; border-collapse: collapse;"> <tr> <td style="width:10%;"></td> <td style="width:10%; text-align:center;">STS062</td> <td style="width:10%; text-align:center;">STS061</td> <td style="width:10%; text-align:center;">STS060</td> <td style="width:70%;">チャンネル6のスタート・トリガ, キャプチャ・トリガの設定</td> </tr> <tr> <td></td> <td style="text-align:center;">0</td> <td style="text-align:center;">0</td> <td style="text-align:center;">0</td> <td>ソフトウェア・トリガ・スタートのみ有効 (他のトリガ要因を非選択にする)</td> </tr> <tr> <td></td> <td style="text-align:center;">0</td> <td style="text-align:center;">0</td> <td style="text-align:center;">1</td> <td>TI06端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジを, スタート・トリガ, キャプチャ・トリガの両方に使用</td> </tr> <tr> <td></td> <td style="text-align:center;">0</td> <td style="text-align:center;">1</td> <td style="text-align:center;">0</td> <td>TI06端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の両エッジを, スタート・トリガとキャプチャ・トリガに分けて使用</td> </tr> <tr> <td></td> <td style="text-align:center;">1</td> <td style="text-align:center;">0</td> <td style="text-align:center;">0</td> <td>マスタ・チャンネルの割り込み信号を使用 (連動動作機能のスレーブ・チャンネル時)</td> </tr> <tr> <td></td> <td colspan="3" style="text-align:center;">上記以外</td> <td>設定禁止</td> </tr> <tr> <td></td> <td style="text-align:center;">MASTER06</td> <td colspan="4">チャンネル6の単体動作機能, 連動動作機能のスレーブ/連動動作機能のマスタの選択</td> </tr> <tr> <td></td> <td style="text-align:center;">0</td> <td colspan="4">単体動作機能, または連動動作機能でスレーブ・チャンネルとして動作</td> </tr> <tr> <td></td> <td style="text-align:center;">1</td> <td colspan="4">連動動作機能でマスタ・チャンネルとして動作</td> </tr> <tr> <td colspan="5">偶数チャンネルのみマスタ・チャンネル (MASTER06 = 1) に設定できます。 奇数チャンネルは必ずスレーブ・チャンネル (MASTER06 = 0) で使用してください。 また, 単体動作機能として使用するチャンネルは, MASTER06 = 0 にします。</td> </tr> <tr> <td></td> <td style="text-align:center;">CCS06</td> <td colspan="4">チャンネル6のカウント・クロック (TCLK) の選択</td> </tr> <tr> <td></td> <td style="text-align:center;">0</td> <td colspan="4">CKS06ビットで指定した動作クロックMCK</td> </tr> <tr> <td></td> <td style="text-align:center;">1</td> <td colspan="4">TI06端子からの入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジ (使用するタイマ入力はTIS0レジスタで選択)</td> </tr> <tr> <td colspan="5">カウント・クロック (TCLK) は, タイマ・カウンタ, 出力制御回路, 割り込み制御回路に使用されます。</td> </tr> <tr> <td></td> <td style="text-align:center;">CKS06</td> <td colspan="4">チャンネル6の動作クロック (MCK) の選択</td> </tr> <tr> <td></td> <td style="text-align:center;">0</td> <td colspan="4">TPS0レジスタで設定した動作クロックCK00</td> </tr> <tr> <td></td> <td style="text-align:center;">1</td> <td colspan="4">TPS0レジスタで設定した動作クロックCK01</td> </tr> <tr> <td colspan="5">動作クロックMCKは, エッジ検出回路に使用されます。また, CCS06ビットの設定によりカウント・クロック (TCLK) を生成します。</td> </tr> </table>																	STS062	STS061	STS060	チャンネル6のスタート・トリガ, キャプチャ・トリガの設定		0	0	0	ソフトウェア・トリガ・スタートのみ有効 (他のトリガ要因を非選択にする)		0	0	1	TI06端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジを, スタート・トリガ, キャプチャ・トリガの両方に使用		0	1	0	TI06端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の両エッジを, スタート・トリガとキャプチャ・トリガに分けて使用		1	0	0	マスタ・チャンネルの割り込み信号を使用 (連動動作機能のスレーブ・チャンネル時)		上記以外			設定禁止		MASTER06	チャンネル6の単体動作機能, 連動動作機能のスレーブ/連動動作機能のマスタの選択					0	単体動作機能, または連動動作機能でスレーブ・チャンネルとして動作					1	連動動作機能でマスタ・チャンネルとして動作				偶数チャンネルのみマスタ・チャンネル (MASTER06 = 1) に設定できます。 奇数チャンネルは必ずスレーブ・チャンネル (MASTER06 = 0) で使用してください。 また, 単体動作機能として使用するチャンネルは, MASTER06 = 0 にします。						CCS06	チャンネル6のカウント・クロック (TCLK) の選択					0	CKS06ビットで指定した動作クロックMCK					1	TI06端子からの入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジ (使用するタイマ入力はTIS0レジスタで選択)				カウント・クロック (TCLK) は, タイマ・カウンタ, 出力制御回路, 割り込み制御回路に使用されます。						CKS06	チャンネル6の動作クロック (MCK) の選択					0	TPS0レジスタで設定した動作クロックCK00					1	TPS0レジスタで設定した動作クロックCK01				動作クロックMCKは, エッジ検出回路に使用されます。また, CCS06ビットの設定によりカウント・クロック (TCLK) を生成します。				
	STS062	STS061	STS060	チャンネル6のスタート・トリガ, キャプチャ・トリガの設定																																																																																																														
	0	0	0	ソフトウェア・トリガ・スタートのみ有効 (他のトリガ要因を非選択にする)																																																																																																														
	0	0	1	TI06端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジを, スタート・トリガ, キャプチャ・トリガの両方に使用																																																																																																														
	0	1	0	TI06端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の両エッジを, スタート・トリガとキャプチャ・トリガに分けて使用																																																																																																														
	1	0	0	マスタ・チャンネルの割り込み信号を使用 (連動動作機能のスレーブ・チャンネル時)																																																																																																														
	上記以外			設定禁止																																																																																																														
	MASTER06	チャンネル6の単体動作機能, 連動動作機能のスレーブ/連動動作機能のマスタの選択																																																																																																																
	0	単体動作機能, または連動動作機能でスレーブ・チャンネルとして動作																																																																																																																
	1	連動動作機能でマスタ・チャンネルとして動作																																																																																																																
偶数チャンネルのみマスタ・チャンネル (MASTER06 = 1) に設定できます。 奇数チャンネルは必ずスレーブ・チャンネル (MASTER06 = 0) で使用してください。 また, 単体動作機能として使用するチャンネルは, MASTER06 = 0 にします。																																																																																																																		
	CCS06	チャンネル6のカウント・クロック (TCLK) の選択																																																																																																																
	0	CKS06ビットで指定した動作クロックMCK																																																																																																																
	1	TI06端子からの入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジ (使用するタイマ入力はTIS0レジスタで選択)																																																																																																																
カウント・クロック (TCLK) は, タイマ・カウンタ, 出力制御回路, 割り込み制御回路に使用されます。																																																																																																																		
	CKS06	チャンネル6の動作クロック (MCK) の選択																																																																																																																
	0	TPS0レジスタで設定した動作クロックCK00																																																																																																																
	1	TPS0レジスタで設定した動作クロックCK01																																																																																																																
動作クロックMCKは, エッジ検出回路に使用されます。また, CCS06ビットの設定によりカウント・クロック (TCLK) を生成します。																																																																																																																		

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

図4-3 タイマ・モード・レジスタ06 (TMR06) のフォーマット (2/2)

CKS06	0	0	CCS06	MAST ER06	STS062	STS061	STS060	CIS061	CIS060	0	0	MD063	MD062	MD061	MD060
-------	---	---	-------	-----------	--------	--------	--------	--------	--------	---	---	-------	-------	-------	-------

動作モード (MD063-MD061で設定)		MD060	カウント・スタートと割り込みの設定				
・インターバル・タイマ・モード	0		カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。				
	1		カウント開始時にタイマ割り込みを発生する (タイマ出力も変化させる)。				
・イベント・カウンタ・モード	0		カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。				
・ワンカウント・モード	0		カウント動作中のスタート・トリガは無効とする。その際に割り込みも発生しない。				
	1		カウント動作中のスタート・トリガを有効とする ^注 。その際に割り込みも発生する。				
・キャプチャ&ワンカウント・モード	0		カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。カウント動作中のスタート・トリガは無効とする。その際に割り込みも発生しない。				
上記以外			設定禁止				
MD063	MD062	MD061	MD060	チャンネル6の動作モードの設定		TCRのカウント動作	単体動作
0	0	0	1/0	インターバル・タイマ・モード		ダウン・カウント	可
0	1	0	1/0	キャプチャ・モード		アップ・カウント	可
0	1	1	0	イベント・カウンタ・モード		ダウン・カウント	可
1	0	0	1/0	ワンカウント・モード		ダウン・カウント	不可
1	1	0	0	キャプチャ&ワンカウント・モード		アップ・カウント	可
上記以外			設定禁止				
CIS061	CIS060	T106端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジ選択 (使用するタイマ入力はTIS0レジスタで選択)					
0	0	立ち下がりエッジ					
0	1	立ち上がりエッジ					
1	0	両エッジ (ロウ・レベル幅測定時) スタート・トリガ: 立ち下がりエッジ, キャプチャ・トリガ: 立ち上がりエッジ					
1	1	両エッジ (ハイ・レベル幅測定時) スタート・トリガ: 立ち上がりエッジ, キャプチャ・トリガ: 立ち下がりエッジ					
STS062-STS060ビット = 010B時以外で両エッジ指定を使用する場合は, CIS061-CIS060ビット = 10Bに設定してください							

注 動作中にスタート・トリガ (TS06 = 1) が掛かると, カウンタをクリアし, 割り込みを発生して再カウント・スタートします。

注意 ビット14, 13, 5, 4には, 必ず0を設定してください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

図4-4 タイマ・モード・レジスタ07 (TMR07) のフォーマット (1/2)

CKS07	0	0	CCS07	MAST ER07	STS072	STS071	STS070	CIS071	CIS070	0	0	MD073	MD072	MD071	MD070
STS072			STS071			STS070			チャンネル7のスタート・トリガ、キャプチャ・トリガの設定						
0			0			0			ソフトウエア・トリガ・スタートのみ有効(他のトリガ要因を非選択にする)						
0			0			1			TI07端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジを, スタート・トリガ, キャプチャ・トリガの両方に使用						
0			1			0			TI07端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の両エッジを, スタート・トリガとキャプチャ・トリガに分けて使用						
1			0			0			マスタ・チャンネルの割り込み信号を使用 (連動動作機能のスレーブ・チャンネル時)						
上記以外			上記以外			上記以外			設定禁止						
MASTER07		チャンネル7の単体動作機能, 連動動作機能のスレーブ/連動動作機能のマスタの選択													
0		単体動作機能, または連動動作機能でスレーブ・チャンネルとして動作													
1		連動動作機能でマスタ・チャンネルとして動作													
偶数チャンネルのみマスタ・チャンネル (MASTER07 = 1) に設定できます。 奇数チャンネルは必ずスレーブ・チャンネル (MASTER07 = 0) で使用してください。 また, 単体動作機能として使用するチャンネルは, MASTER07 = 0 にします。															
CCS07		チャンネル7のカウンタ・クロック (TCLK) の選択													
0		CKS07ビットで指定した動作クロックMCK													
1		TI07端子からの入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジ (使用するタイマ入力はTIS0レジスタで選択)													
カウンタ・クロック (TCLK) は, タイマ・カウンタ, 出力制御回路, 割り込み制御回路に使用されます。															
CKS07		チャンネル7の動作クロック (MCK) の選択 ^注													
0		TPS0レジスタで設定した動作クロックCK00													
1		TPS0レジスタで設定した動作クロックCK01													
動作クロックMCKは, エッジ検出回路に使用されます。また, CCS07ビットの設定によりカウンタ・クロック (TCLK) を生成します。															

注 マスタ・チャンネルと同一の設定にしてください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

図4-4 タイマ・モード・レジスタ07 (TMR07) のフォーマット (2/2)

CKS 07	0	0	CCS 07	MAST ER07	STS 072	STS 071	STS 070	CIS 071	CIS 070	0	0	MD 073	MD 072	MD 071	MD 070
-----------	---	---	-----------	--------------	------------	------------	------------	------------	------------	---	---	-----------	-----------	-----------	-----------

動作モード (MD073-MD071で設定)		MD070	カウント・スタートと割り込みの設定				
・インターバル・タイマ・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。				
	1	1	カウント開始時にタイマ割り込みを発生する (タイマ出力も変化させる)。				
・イベント・カウンタ・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。				
・ワンカウント・モード	0	0	カウント動作中のスタート・トリガは無効とする。 その際に割り込みも発生しない。				
	1	1	カウント動作中のスタート・トリガを有効とする ^注 。 その際に割り込みも発生する。				
・キャプチャ&ワンカウント・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。 カウント動作中のスタート・トリガは無効とする。 その際に割り込みも発生しない。				
上記以外			設定禁止				
MD073	MD072	MD071	MD070	チャンネル7の動作モードの設定		TCRのカウント動作	単体動作
0	0	0	1/0	インターバル・タイマ・モード		ダウン・カウント	可
0	1	0	1/0	キャプチャ・モード		アップ・カウント	可
0	1	1	0	イベント・カウンタ・モード		ダウン・カウント	可
1	0	0	1/0	ワンカウント・モード		ダウン・カウント	不可
1	1	0	0	キャプチャ&ワンカウント・モード		アップ・カウント	可
上記以外				設定禁止			
CIS071	CIS070	TI07端子入力信号, $f_{SUB}/2$, $f_{SUB}/4$, またはINTRTC1の有効エッジ選択 (使用するタイマ入力はTIS0レジスタで選択)					
0	0	立ち下がりエッジ					
0	1	立ち上がりエッジ					
1	0	両エッジ (ロウ・レベル幅測定時) スタート・トリガ: 立ち下がりエッジ, キャプチャ・トリガ: 立ち上がりエッジ					
1	1	両エッジ (ハイ・レベル幅測定時) スタート・トリガ: 立ち上がりエッジ, キャプチャ・トリガ: 立ち下がりエッジ					
STS072-STS070ビット = 010B時以外で両エッジ指定を使用する場合は, CIS071-CIS070ビット = 10Bに設定してください							

注 動作中にスタート・トリガ (TS07 = 1) が掛かると, カウンタをクリアし, 割り込みを発生して再カウント・スタートします。

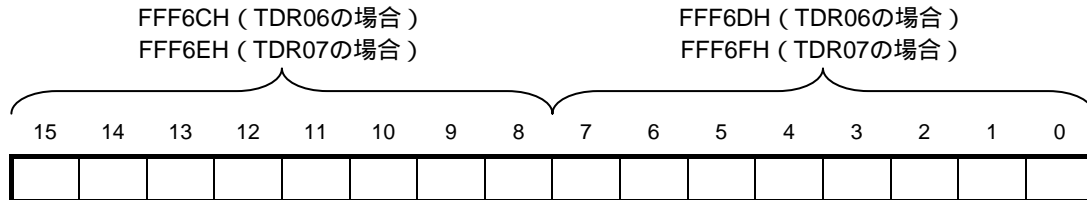
注意 ビット14, 13, 5, 4には, 必ず0を設定してください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

(3) タイマ・データ・レジスタ06, 07 (TDR06, TDR07)

キャプチャ機能とコンペア機能を切り替えて使用できる16ビットのレジスタです。このサンプル・プログラムでは、TDR06およびTDR07をコンペア・レジスタとして使用しますので、TDR06およびTDR07に設定した値からダウン・カウントをスタートして、0000Hになったときに割り込み信号(INTTM06 ,INTTM07)を発生します。TDR06およびTDR07は書き換えられるまで値を保持します。

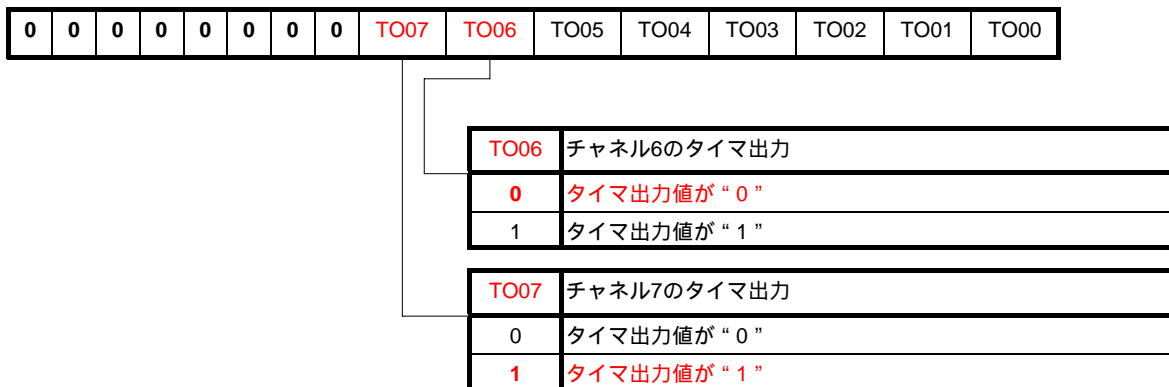
図4 - 5 タイマ・データ・レジスタ06, 07 (TDR06, TDR07) のフォーマット



(4) タイマ出力レジスタ0 (TO0)

TO0は、各チャンネルのタイマ出力のバッファ・レジスタです。
このレジスタの各ビットの値が、各チャンネルのタイマ出力端子 (TO0n) から出力されます。
このレジスタのTO0のソフトウェアによる書き換えは、タイマ出力禁止時 (TOE0n = 0) のみ可能です。
タイマ出力許可時 (TOE0n = 1) は、ソフトウェアによる書き換えは無視され、タイマ動作によってのみ値が変更されます。

図4 - 6 タイマ出力レジスタ0 (TO0) のフォーマット



注意 ビット15-8には必ず0を設定してください。

備考1. n = 7-0

2. 図の赤字部分がサンプル・プログラムでの設定値となります。

(5) タイマ出力許可レジスタ0 (TOE0)

TOE0は、各チャンネルのタイマ出力許可 / 禁止を設定するレジスタです。

タイマ出力を許可したチャンネルnは、後述のタイマ出力レジスタ (TO0) のTO0nの値をソフトウェアによって書き換えできなくなり、カウント動作によるタイマ出力機能によって反映された値がタイマ出力端子 (TO0n) から出力されます。

図4 - 7 タイマ出力許可レジスタ0 (TOE0) のフォーマット



注意 ビット15-8には必ず0を設定してください。

備考1 . n = 7-0

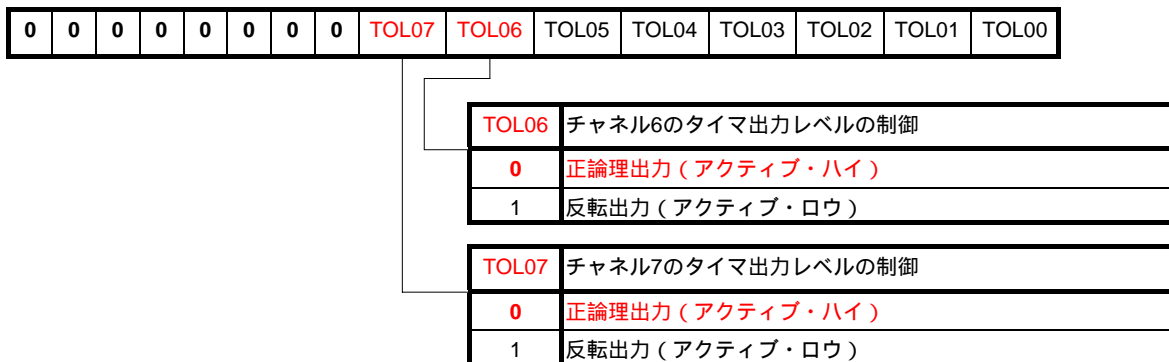
2. 図の赤字部分がサンプル・プログラムでの設定値となります。

(6) タイマ出力レベル・レジスタ0 (TOL0)

TOL0は、各チャンネルのタイマ出力レベルを制御するレジスタです。

このレジスタによる各チャンネルnの反転設定は、タイマ出力許可 (TOE0n = 1)、連動動作モード (TOM0n = 1) 時にタイマ出力信号がセット、リセットされるタイミングで反映されます。トグル・モード (TOM0n = 0) 時には、このレジスタの設定は無効となります。

図4 - 8 タイマ出力レベル・レジスタ0 (TOL0) のフォーマット



(注意と備考は次ページにあります。)

注意 ビット15-8には必ず0を設定してください。

備考1. n = 7-0

2. タイマ動作中にこのレジスタの値を書き換えた場合、書き換えた直後のタイミングではなく、次にタイマ出力信号が変化するタイミングで、タイマ出力が反転出力になります。
3. 図の赤字部分がサンプル・プログラムでの設定値となります。

(7) タイマ出力モード・レジスタ0 (TOM0)

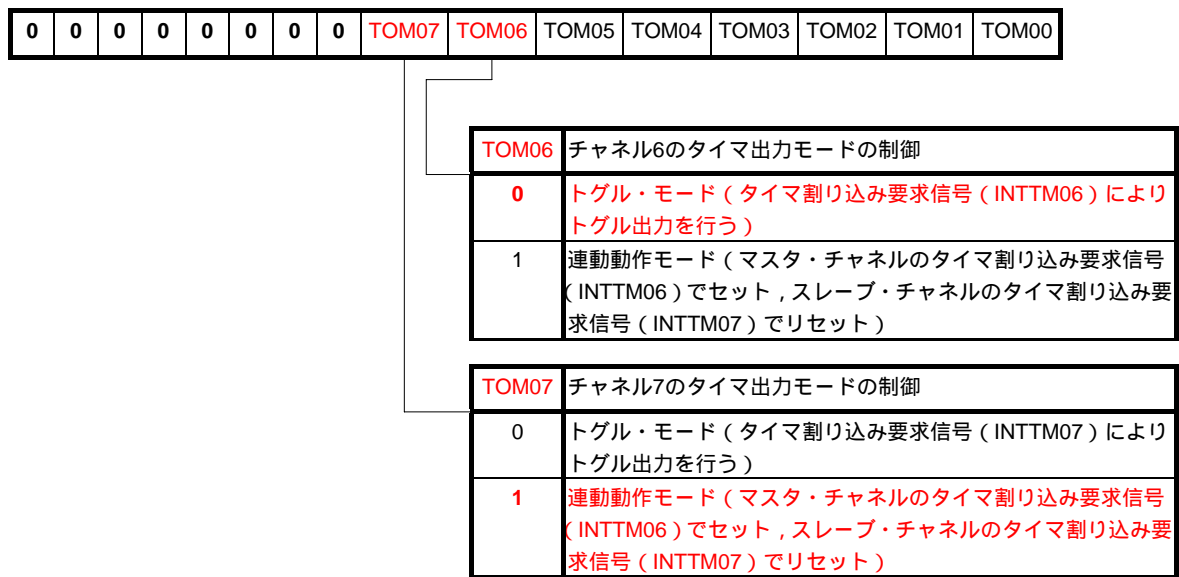
TOM0は、各チャンネルのタイマ出力モードを制御するレジスタです。

単体動作機能として使用する場合、使用するチャンネルの対応ビットを0に設定します。

連動動作機能 (PWM出力、ワンショット・パルス出力、多重PWM出力)として使用する場合、マスタ・チャンネルの対応ビットを0に設定し、スレーブ・チャンネルの対応ビットを1に設定します。

このレジスタによる各チャンネルnの設定は、タイマ出力許可 (TOE0n = 1)時にタイマ出力信号がセット、リセットされるタイミングで反映されます。

図4 - 9 タイマ出力モード・レジスタ0 (TOM0) のフォーマット



注意 ビット15-8には必ず0を設定してください。

備考1. n = 7-0

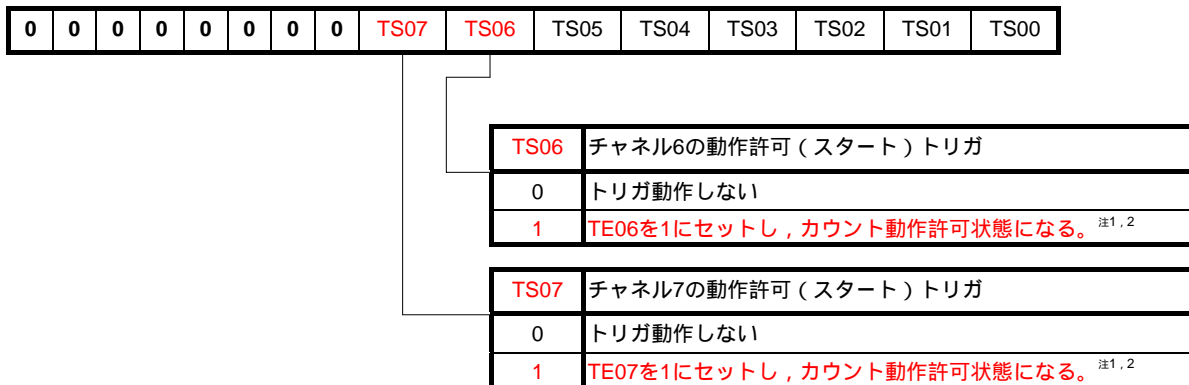
2. 図の赤字部分がサンプル・プログラムでの設定値となります。

(8) タイマ・チャンネル開始レジスタ0 (TS0)

TS0は、タイマ・カウンタ (TCR0n) をクリアし、カウント動作の開始を各チャンネルごとに設定するトリガ・レジスタです。

各ビット (TS0n) が1にセットされると、タイマ・チャンネル許可ステータス・レジスタ0 (TE0) の対応ビット (TE0n) が1にセットされます。TS0nはトリガ・ビットなので、TE0n = 1になるとすぐTS0nはクリアされます。

図4 - 10 タイマ・チャンネル開始レジスタ0 (TS0) のフォーマット



- 注1. インターバル・タイマ・モードの場合、スタート・トリガ検出 (TS0n = 1) 後、カウント・クロック発生まで何も動作しません。最初のカウント・クロックでTDR0nの値をTCR0nにロードし、以降のカウント・クロックでダウン・カウント動作を行います。
2. 連動させるチャンネルを同時スタートさせるため、連動させるチャンネルのTS0nビットを同時に設定する必要があります。

注意 ビット15-8には必ず0を設定してください。

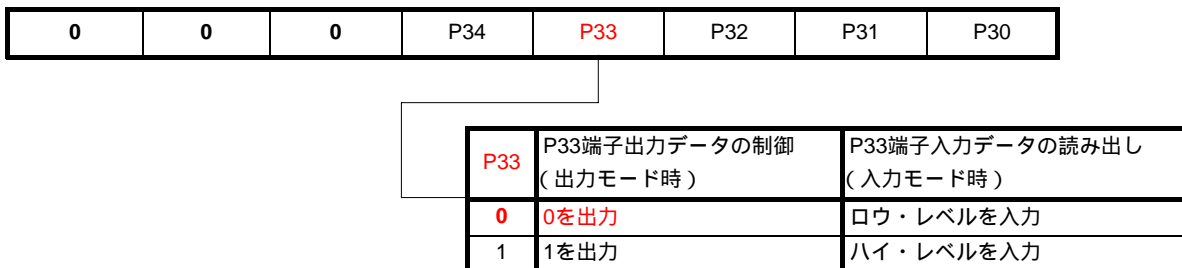
備考1. n = 7-0

2. TS0レジスタの読み出し値は常に0となります。
3. 図の赤字部分がサンプル・プログラムでの設定値となります。

(9) ポート・レジスタ3 (P3)

ポート出力時にチップ外に出力するデータをライトするレジスタです。P33端子をPWM出力として使用する場合は、P33に0を設定してください。

図4 - 11 ポート・レジスタ3 (P3) のフォーマット



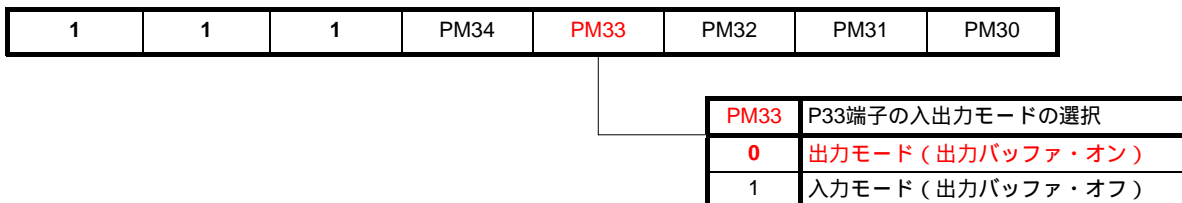
注意 ビット7-5には必ず0を設定してください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

(10) ポート・モード・レジスタ3 (PM3)

ポートの入力 / 出力を1ビット単位で設定するレジスタです。P33端子をタPWM出力として使用する場合は、PM33に0を設定してください。

図4 - 12 ポート・モード・レジスタ3 (PM3) のフォーマット



注意 ビット7-5には必ず1を設定してください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

4.3 ボルテージ・リファレンスの設定

ボルテージ・リファレンスは、次のレジスタを使用します。

- ・アナログ基準電圧制御レジスタ (ADVRC)

【温度測定を行う場合のボルテージ・リファレンスの設定手順例】

- 周辺イネーブル・レジスタ0 (PER0) のビット5 (ADCEN) をセット (1) (4.1参照)
- アナログ基準電圧制御レジスタ (ADVRC) のビット3 (VRSEL) をセット (1)
- アナログ基準電圧制御レジスタ (ADVRC) のビット1, 0 (VRGV, VRON) をセット (1)
- ボルテージ・リファレンスの動作安定待ちのため、ソフトウェアで17ms以上ウエイト

(1) アナログ基準電圧制御レジスタ (ADVRC)

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

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

図4 - 13 アナログ基準電圧制御レジスタ (ADVRC) のフォーマット

ADREF	0	0	0	VRSEL	0	VRGV	VRON
-------	---	---	---	-------	---	------	------

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

ADREF	A/Dコンバータの-側の基準電圧源の選択
0	AV _{SS}
1	AV _{REFM} (外部基準電圧入力)

(注意と備考は次ページにあります。)

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

- 備考1. 図の赤字部分がサンプル・プログラムでの設定値となります。
2. ADREF，VRSELとVRONの設定により、選択可能な+側と-側の基準電圧源の組み合わせは次のようになります。

表4 - 1 ADREF，VRSELとVRONの設定

ADEF	VRSEL	VRON	A/Dコンバータの+側の基準電圧（ AD_{REFP} ）	A/Dコンバータの-側の基準電圧（ AD_{REFM} ）
0	0	0	AV_{REFP}	AV_{SS}
0	1	1	V_{REFOUT} （VR出力）	AV_{SS}
1	0	0	AV_{REFP}	AV_{REFM}
1	1	1	V_{REFOUT} （VR出力）	AV_{REFM}

4.4 A/Dコンバータの設定

A/Dコンバータは、次の6種類のレジスタを使用します。

- ・周辺イネーブル・レジスタ0 (PER0)
- ・A/Dコンバータ・モード・レジスタ (ADM)
- ・A/Dコンバータ・モード・レジスタ1 (ADM1)
- ・アナログ入力チャンネル指定レジスタ (ADS)
- ・A/Dポート・コンフィギュレーション・レジスタ (ADPC)
- ・ポート・モード・レジスタ2 (PM2)

【温度測定を行う場合のA/Dコンバータの設定手順例】

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

ポート・モード・レジスタ2 (PM2) でP21端子を入力モードに設定

A/Dコンバータ・モード・レジスタ (ADM) で動作モードをシングル変換モード, A/D変換時間を12 μ sに設定

A/Dコンバータ・モード・レジスタ (ADM) でA/D電圧コンパレータを動作許可に設定

A/Dポート・コンフィギュレーション・レジスタ (ADPC) でANI1をアナログ入力に設定

アナログ入力チャンネル指定レジスタ (ADS) でアナログ入力チャンネルにANI1を選択

A/Dコンバータ・モード・レジスタ1 (ADM1) でA/Dのトリガ・モードをソフトウェア・トリガ・モードに設定

A/Dコンバータ・モード・レジスタ (ADM) でA/D変換動作を開始

(1) A/Dコンバータ・モード・レジスタ (ADM)

A/D変換するアナログ入力の変換時間，変換動作の開始 / 停止を設定するレジスタです

図4 - 14 A/Dコンバータ・モード・レジスタ (ADM) のフォーマット

ADCS	ADSCM	FR2 ^{注1}	FR1 ^{注1}	FR0 ^{注1}	LV1 ^{注1}	LV0 ^{注1}	ADCE						
						<table border="1"> <tr> <td>ADCE</td> <td>A/D電圧コンパレータの動作制御^{注2}</td> </tr> <tr> <td>0</td> <td>A/D電圧コンパレータの動作停止</td> </tr> <tr> <td>1</td> <td>A/D電圧コンパレータの動作許可</td> </tr> </table>		ADCE	A/D電圧コンパレータの動作制御 ^{注2}	0	A/D電圧コンパレータの動作停止	1	A/D電圧コンパレータの動作許可
ADCE	A/D電圧コンパレータの動作制御 ^{注2}												
0	A/D電圧コンパレータの動作停止												
1	A/D電圧コンパレータの動作許可												
						<table border="1"> <tr> <td>ADSCM</td> <td>A/D変換動作モードを指定</td> </tr> <tr> <td>0</td> <td>連続変換モード</td> </tr> <tr> <td>1</td> <td>シングル変換モード</td> </tr> </table>		ADSCM	A/D変換動作モードを指定	0	連続変換モード	1	シングル変換モード
ADSCM	A/D変換動作モードを指定												
0	連続変換モード												
1	シングル変換モード												
						<table border="1"> <tr> <td>ADCS</td> <td>A/D変換動作の制御^{注2,3,4}</td> </tr> <tr> <td>0</td> <td>変換動作停止</td> </tr> <tr> <td>1</td> <td>変換動作許可</td> </tr> </table>		ADCS	A/D変換動作の制御 ^{注2,3,4}	0	変換動作停止	1	変換動作許可
ADCS	A/D変換動作の制御 ^{注2,3,4}												
0	変換動作停止												
1	変換動作許可												

- 注1. FR2-FR0, LV1, LV0およびA/D変換に関する詳細については，表4 - 3 A/D変換時間の選択を参照してください。
2. A/D電圧コンパレータはADCSとADCEで動作制御され，動作開始から安定するまでに，1μs (T.B.D) かかります。このため，ADCEに1を設定してから1μs以上 (T.B.D) 経過したあとに，ADCSに1を設定することで，最初の変換データより有効となります。1μs以上 (T.B.D) ウェイトしないでADCSに1を設定した場合は，最初の変換データを無視してください。
3. タイマ・トリガ・モードで使用する場合，ADCSに1を設定しないでください (タイマ・トリガ信号発生時に，ADCSは自動的に1に切り替わります)。ただし，A/D変換動作を停止するために，ADCSに0を設定することは有効です。
4. ADCSをリードすることにより，A/D変換動作中かどうかを知ることができます。

- 注意1. ADSCM, FR2-FR0, LV1, LV0を同一データ以外に書き換える場合は，いったんA/D変換動作を停止させたのちに行ってください。
2. 変換モード2 (LV1 = 0, LV0 = 1) または変換モード3 (LV1 = 1, LV0 = 0) で使用する場合は，アナログ基準電圧制御レジスタ (ADVRC) でA/Dコンバータ用入力ゲート昇圧回路を動作許可に設定してから，ADCEとADCSに1を設定してください。このとき，A/Dコンバータ用入力ゲート昇圧回路を動作許可に設定してから，昇圧回路動作安定待ち時間 (T.B.D) を経過したあとに，ADCSに1を設定してください。
3. アナログ入力ソースとして使用する場合は，A/D変換動作設定前に，オペアンプの動作を開始してください (78K0R/Lx3 ユーザーズ・マニュアルを参照)。また，A/D変換動作中に，オペアンプの設定を変更しないでください。
4. ボルテージ・リファレンスの出力電圧をA/Dコンバータの+側の基準電圧 (AD_{REFP}) として使用する場合は，A/D変換動作設定前に，ボルテージ・リファレンスの動作を開始してください (78K0R/Lx3 ユーザーズ・マニュアルを参照)。また，A/D変換動作中に，ボルテージ・リファレンスの設定を変更しないでください。

(備考は次ページにあります。)

備考1. 図の赤字部分がサンプル・プログラムでの設定値となります。

- ADREF, VRSELとVRONの設定により, 選択可能な+側と-側の基準電圧源の組み合わせは次のようになります。

表4-2 ADCSとADCEの設定

ADCS	ADCE	A/D変換動作
0	0	停止状態 (DC電力消費パスは存在しません)
0	1	変換待機モード (A/D電圧コンパレータ動作, コンパレータのみ電力消費)
1	0	設定禁止
1	1	変換モード (A/D電圧コンパレータ動作)

表4-3 A/D変換時間の選択

A/Dコンバータ・モード・レジスタ (ADM)					モード	変換時間の選択		変換クロック (f_{AD})
FR2	FR1	FR0	LV1	LV0		$f_{CLK} = 20 \text{ MHz}$		
0	0	0	0	0	変換モード1 ^{注1}	240/ f_{CLK}	12 μs	$f_{CLK}/12$
0	0	1				160/ f_{CLK}	8 μs	$f_{CLK}/8$
0	1	0				120/ f_{CLK}	6 μs	$f_{CLK}/6$
0	1	1				100/ f_{CLK}	5 μs	$f_{CLK}/5$
1	0	0				80/ f_{CLK}	設定禁止	$f_{CLK}/4$
1	0	1				60/ f_{CLK}		$f_{CLK}/3$
1	1	0				40/ f_{CLK}		$f_{CLK}/2$
1	1	1				20/ f_{CLK}		f_{CLK}
0	0	0				0		1
0	0	1	160/ f_{CLK}	8 μs	$f_{CLK}/8$			
0	1	0	120/ f_{CLK}	6 μs	$f_{CLK}/6$			
0	1	1	100/ f_{CLK}	5 μs	$f_{CLK}/5$			
1	0	1	80/ f_{CLK}	設定禁止	$f_{CLK}/4$			
1	1	0	60/ f_{CLK}		$f_{CLK}/3$			
1	1	1	40/ f_{CLK}		$f_{CLK}/2$			
1	1	1	20/ f_{CLK}		f_{CLK}			
0	0	0	1		1		変換モード3 ^{注3}	
0	0	1		200/ f_{CLK}		10 μs		$f_{CLK}/8$
0	1	0		150/ f_{CLK}		7.5 μs		$f_{CLK}/6$
0	1	1		125/ f_{CLK}		6.25 μs		$f_{CLK}/5$
1	0	1		100/ f_{CLK}		設定禁止		$f_{CLK}/4$
1	1	0		75/ f_{CLK}				$f_{CLK}/3$
1	1	1		50/ f_{CLK}				$f_{CLK}/2$
1	1	1		25/ f_{CLK}				f_{CLK}
上記以外								設定禁止

注1. 変換モード1: 2.7 V AV_{DD0} 5.5 V, A/Dコンバータ用入力ゲート昇圧回路動作停止時

2. 変換モード2: 2.3 V AV_{DD0} 5.5 V, A/Dコンバータ用入力ゲート昇圧回路動作時

3. 変換モード3: 1.8 V AV_{DD0} 5.5 V, A/Dコンバータ用入力ゲート昇圧回路動作時

備考1. f_{CLK} : CPU / 周辺ハードウェア・クロック周波数

- 表の赤字部分がサンプル・プログラムでの設定値となります。

(2) A/Dコンバータ・モード・レジスタ1 (ADM1)

A/D変換起動トリガを設定するレジスタです。

図4 - 15 A/Dコンバータ・モード・レジスタ1 (ADM1) のフォーマット

ADTMD	0	0	0	0	0	0	ADTRS
-------	---	---	---	---	---	---	-------

ADTMD	A/Dトリガ・モードの選択
0	ソフトウェア・トリガ・モード
1	タイマ・トリガ・モード (ハードウェア・トリガ・モード)

注意1. ビット6-1には、必ず0を設定してください。

2. A/D変換中にADM1を書き換えることは禁止です。変換動作停止時(ADCS = 0)に書き換えてください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

(3) アナログ入力チャンネル指定レジスタ (ADS)

ADSレジスタは、A/Dコンバータのアナログ入力チャンネルを指定するレジスタです。オペアンプのAMP00端子はA/Dコンバータのアナログ入力端子ANI1と兼用しているため、ANI1をA/Dコンバータのアナログ入力チャンネルに設定することで、オペアンプのシングル・アンプ・モードで増幅したアナログ電圧をA/Dコンバータのアナログ入力として使用することができます。

図4 - 16 アナログ入力チャンネル指定レジスタ (ADS) のフォーマット

0	0	0	0	ADS3	ADS2	ADS1	ADS0
---	---	---	---	------	------	------	------

ADS3	ADS2	ADS1	ADS0	アナログ入力チャンネル
0	0	0	0	ANI0
0	0	0	1	ANI1
0	0	1	0	ANI2
0	0	1	1	ANI3
0	1	0	0	ANI4
0	1	0	1	ANI5
0	1	1	0	ANI6
0	1	1	1	ANI7
1	0	0	0	ANI8
1	0	0	1	ANI9
1	0	1	0	ANI10
1	1	1	1	ANI15
上記以外				設定禁止

(注意と備考は次ページにあります。)

- 注意1. ビット7-4には必ず0を設定してください。
- A/D変換で使用するチャンネルは、ポート・モード・レジスタ2, 15 (PM2, PM15) で入力モードに選択してください。
 - ADPCでデジタル入出力として設定する端子を、ADSで設定しないでください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

(4) A/Dポート・コンフィギュレーション・レジスタ (ADPC)

ANI0/AMP0-/P20-ANI7/AMP2O/P27, ANI8/AMP2+/P150-ANI10/P152, ANI15/AV_{REFM}/P157端子を、アナログ入力/ポートのデジタル入出力に切り替えるレジスタです。A/D変換で使用する端子は、アナログ入力に設定してください。

図4 - 17 A/Dポート・コンフィギュレーション・レジスタ (ADPC) のフォーマット

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

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

- 注意1. A/D変換で使用するチャンネルは、ポート・モード・レジスタ2, 15 (PM2, PM15) で入力モードに選択してください。
- ビット7-5には必ず0を設定してください。

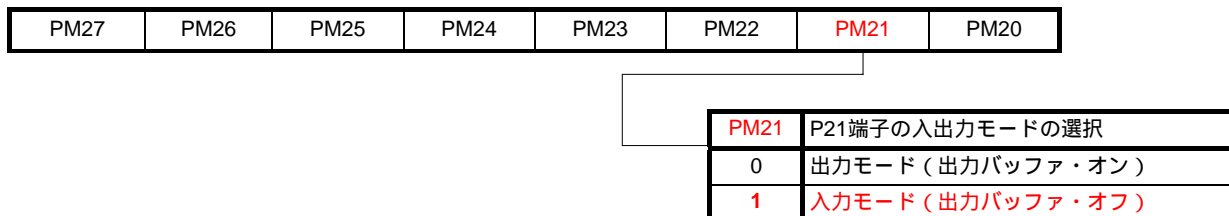
備考 図の赤字部分がサンプル・プログラムでの設定値となります。

(5) ポート・モード・レジスタ2 (PM2)

ANI1/AMP00/P21端子をアナログ入力ポートとして使用するとき、PM21に1を設定してください。このときP21の出力ラッチは、0または1のどちらでもかまいません。

PM21に0を設定した場合は、アナログ入力ポートとして使用することはできません。

図4 - 18 ポート・モード・レジスタ2 (PM2) のフォーマット



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

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

ANI1/AMP00/P21端子の機能は、ADPCレジスタ、ADSレジスタ、PM2レジスタ、OAEN0ビット、ADREFビットの設定で決定します。

表4 - 4 ANI1/AMP00/P21端子機能の設定

ADPC レジスタ	PM2, レジスタ	OAEN0 ビット	ADSレジスタ	ANI1/AMP00/P21 端子機能の設定
デジタル入 出力選択	入力モード	0	-	デジタル入力
		1	-	設定禁止
	出力モード	0	-	デジタル出力
		1	-	設定禁止
アナログ入 力選択	入力モード	0	ANI選択	アナログ入力 (A/D変換対象)
			ANI非選択	アナログ入力 (A/D変換非対象)
		1	ANI選択	設定禁止
			ANI非選択	オペアンプ入力
出力モード	-	-	設定禁止	

備考 表の赤字部分がサンプル・プログラムでの設定値となります。

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

4.5 ソフトウェア記述例

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

(1) アセンブリ言語

タイマ・アレイ・ユニット0の設定 (PWM出力として使用する場合)

```

MOV PM3, #11100000B ; Set P30-P34 to output port
... (略) ...
SINITTAU0:
MOV TPSOL, #00001000B ; Timer 0
... (略) ...
MOVW AX, #1000100000000001B ; Channel 6
MOVW TMR06, AX
... (略) ...
MOVW AX, #1000010000001001B ; Channel 7
MOVW TMR07, AX
... (略) ...
; CK01 = fCLK = 20MHz -> 0.05[us/clock] * 0xFF
MOVW TDR06, #(0FFFH - 1) ; set interval
MOVW TDR07, #0FFFH ; set interval
MOVW HL, #LOWW TOM0L ; TO06: toggle mode
CLR1 [HL].6 ; TO06: toggle mode
SET1 [HL].7 ; TO07: combination
MOVW HL, #LOWW TOL0L ; TO06: set output active level High
CLR1 [HL].6 ; TO06: set output active level High
CLR1 [HL].7 ; TO07: high
MOVW HL, #LOWW TOOL ; TO06: output disabled
CLR1 [HL].6 ; TO06: output disabled
SET1 [HL].7 ; TO07: output enabled
MOVW HL, #LOWW TOE0L ; TO06: output disabled
CLR1 [HL].6 ; TO06: output disabled
SET1 [HL].7 ; TO07: output enabled
CLR1 P3.3 ; TO07: output latch
SET1 TMMK06 ; TO06: output latch
SET1 TMMK07 ; TO07: output latch
CLR1 TMIF06 ; clear interrupt
MOV TSOL, #0C0H ; timer 0

```

P33をPWM出力端子用に出力モードに設定
 CK00を $f_{CLK}/2^8$, CK01を f_{CLK} に設定
 チャンネル6をPWM出力のマスター・チャンネルとしての動作クロックをCK01, 動作モードをインターバル・タイマ・モードに設定
 チャンネル7をPWM出力のスレーブ・チャンネルとしての動作クロックをCK01, 動作モードをワンカウント・モードに設定
 TDR07が0x0FFFのときPWM出力のデューティ比が100%となるようにパルス周期を設定
 デューティ比の初期設定が100%となるように設定
 チャンネル6をトグル・モード, チャンネル7を連動動作モードに設定
 チャンネル6, 7のタイマ出力レベルを正論理出力(アクティブ・ハイ)に設定
 チャンネル6のタイマ出力値を0に, チャンネル7のタイマ出力値を1に設定
 チャンネル6をカウント動作によるTO06出力動作停止, チャンネル7をカウント動作によるTO07出力動作許可に設定
 TO07端子の出力ラッチを0に設定
 チャンネル6, 7のタイマ割り込みをマスク設定
 チャンネル6, 7の動作許可
 チャンネル6(マスター・チャンネル)の割り込み要求をクリア

ボルテージ・リファレンスの設定

```

MOVW AX, #000000000000000B ; Timer Mode Register 01
MOVW TMR01, AX
... (略) ...
; CK00 = fCLK/2^8 = 78.1kHz -> 12.804[us/clock] * 1328 = 17.003ms
MOVW TDR01, #(1328 - 1) ; set timer period about 17 ms
SET1 TMMK01 ; enable timer
... (略) ...
SINITVR:
MOV ADVRC, #00001000B ; Analog Voltage Reference Register
... (略) ...
SET1 ADVRC.0 ; Enable operation
SET1 ADVRC.1 ; Output 2.0V
; wait for settling time to 100ns
SET1 TSOL.1 ; start TAU0 CH1
CLR1 TMIF01 ; clear interrupt
JINIVR100:
NOP ; wait
BF TMIF01, $JINIVR100 ; and
CLR1 TMIF01 ; clear
SET1 TTOL.1 ; stop TAU0
    
```

タイマ・アレイ・ユニット0のチャネル0の動作クロックをCK01, 動作モードをインターバル・タイマ・モードに設定

タイマ割り込み周期を約17msに設定

のタイマ割り込みをマスク設定

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

ボルテージ・リファレンスの出力電圧に2.0Vを選択

ボルテージ・リファレンスの動作許可

タイマ・アレイ・ユニット0のチャネル0を使用し, ボルテージ・リファレンスの動作安定待ちとして約17msウエイト

A/Dコンバータの設定

```

MOV PM2, #11111110B ; Set P20 to output port, P21-P27 to input port
... (略) ...
SINITADC:
MOV ADM, #01000011B ; A/D Converter Mode Register
... (略) ...
MOV ADPC, #0000001B ; A/D Port Configuration Register
... (略) ...
MOV ADS, #0000001B ; Analog Input Channel Specification Register
... (略) ...
MOV ADM1, #0000000B ; A/D Converter Mode Register 1
... (略) ...
SET1 ADMK ; disable A/D interrupt
    
```

P21端子を入力モードに設定

動作モードをシングル変換モード, A/D変換時間を12μs, A/D電圧コンパレータを動作許可に設定

P21端子をANI1として使用できるようにアナログ入力端子に設定

アナログ入力チャンネルにANI1を選択

A/Dコンバータのトリガ・モードをソフトウェア・トリガ・モードに設定

A/D変換終了割り込みをマスク設定

(2) C言語

タイマ・アレイ・ユニット0の設定 (PWM出力として使用する場合)

```

PM3 = 0b11100000; /* Set P30-P34 to output port */
... (略) ...

static void fn_InitTau0(void)
{
    TPSOL = 0b00001000; /* Timer Clock
    ... (略) ...

    TMR06 = 0b1000100000000001; /* Timer Mode Register 06 */
    ... (略) ...

    TMR07 = 0b1000010000001001; /* Timer Mode Register 07 */
    ... (略) ...

    /* CK01 = fCLK = 20MHz -> 0.05[us/clk] * 0xFFFF[0
    TDR06 = (0x0FFF - 1); /* set int
    TDR07 = 0x0FFF; /* set int

    TOM0L &= 0xbf; /* TM06:togg
    TOM0L |= 0x80; /* TM07:combination
    TOL0L &= 0x3f; /* set output activ level High */
    TO0L &= 0x0bf; /* TO06:Low
    TO0L |= 0x080; /* TO07:
    TOE0L &= 0xbf; /* TO06:
    TOE0L |= 0x80; /* TO07:out

    P3.3 = 0; /* TO07 output low */
    TMMK06 = 1; /* disable int
    TMMK07 = 1; /* d
    TMIF06 = 0; /* clear inte
    TS0L = 0x0c0; /* TAU0 CH6 & CH7 start */

```

P33をPWM出力端子用に出力モードに設定
 CK00を $f_{CLK}/2^8$, CK01を f_{CLK} に設定
 チャンネル6をPWM出力のマスター・チャンネルとしての動作クロックをCK01,動作モードをインターバル・タイム・モードに設定
 チャンネル7をPWM出力のスレーブ・チャンネルとしての動作クロックをCK01,動作モードをワンカウント・モードに設定
 TDR07が0x0FFFのときPWM出力のデューティ比が100%となるようにパルス周期を設定
 デューティ比の初期設定が100%となるように設定
 チャンネル6をトグル・モード,チャンネル7を連動動作モードに設定
 チャンネル6,7のタイマ出力レベルを正論理出力(アクティブ・ハイ)に設定
 チャンネル6のタイマ出力値を0に,チャンネル7のタイマ出力値を1に設定
 チャンネル6をカウント動作によるTO06出力動作停止,チャンネル7をカウント動作によるTO7出力動作許可に設定
 TO07端子の出力ラッチを0に設定
 チャンネル6,7のタイマ割り込みをマスク設定
 チャンネル6(マスター・チャンネル)の割り込み要求をクリア
 チャンネル6,7の動作許可

ボルテージ・リファレンスの設定

```

TMR01 = 0b0000000000000000; /* Timer Mode Register 01 */
... (略) ...
/* CK00 = fCLK/2^8 = 78.1kHz -> 12.804[us] */
TDR01 = (1328 - 1);
TMMK01 = 1;
... (略) ...
static void fn_InitVr(void)
{
ADVRC = 0b00001000; /* Analog Reference Control Register */
... (略) ...
ADVRC.0 = 1; /* Enable VREFOUT operation */
ADVRC.1 = 1; /* Output 2.0V */
/* wait for settling time */
TSOL.1 = 1; /* start TAU0 CH1 */
TMIF01 = 0;
while(!TMIF01){
NOP();
}
TMIF01 = 0;
TTOL.1 = 1; /* stop TAU0 CH1 */

```

タイマ・アレイ・ユニット0のチャンネル0の動作クロックをCK01, 動作モードをインターバル・タイム・モードに設定

タイマ割り込み周期を約17msに設定

のタイマ割り込みをマスク設定

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

ボルテージ・リファレンスの出力電圧に2.0Vを選択

ボルテージ・リファレンスの動作許可

ボルテージ・リファレンスの動作安定待ちとして約17msウエイト

A/Dコンバータの設定

```

PM2 = 0b11111110; /* Set P20 to output port, P21-P27 to input port */
... (略) ...
static void fn_InitAdc(void)
{
ADM = 0b01000011; /* A/D Converter Mode Register */
... (略) ...
ADPC = 0b00000001; /* A/D Port Configuration Register */
... (略) ...
ADS = 0b00000001; /* Analog Input Channel Selection Register */
... (略) ...
ADM1 = 0b00000000; /* A/D Converter Mode Register 1 */
... (略) ...
ADMK = 1; /* A/D Converter Mode Register 2 */

```

P21端子を入力モードに設定

動作モードをシングル変換モード, A/D変換時間を12μs, A/D電圧コンバータを動作許可に設定

P21端子をANI1として使用できるようにアナログ入力端子に設定

アナログ入力チャンネルにANI1を選択

A/Dコンバータのトリガ・モードをソフトウェア・トリガ・モードに設定

A/D変換終了割り込みをマスク設定

4.6 温度測定処理

温度測定処理について説明します。

アセンブリ言語の温度測定処理では、次の動作を行います。

HALTモードに移行し、タイマ・アレイ・ユニット0のチャンネル0による約500ms周期のタイマ割り込み要求信号（INTTM00）の発生を待ちます。タイマ割り込み要求信号（INTTM00）が発生すると、温度測定処理を行います。

A/D変換を行い、温度センサの出力電圧を取得します。

- (a) A/D変換の精度を向上させるため、A/D変換中はHALTモードに移行します。
- (b) A/D変換終了割り込み発生後、12ビットA/D変換結果レジスタ（ADCR）を読み出し、A/D変換結果保存用バッファに保存します。
- (c) リセット後1回目のA/D変換結果を、A/D変換結果保存用バッファの初期値として全ての要素に保存します。
- (d) A/D変換結果保存用バッファのポインタを更新します。

温度測定の精度を向上させるため、A/D変換結果保存用バッファに保存した10回のA/D変換結果から平均値を算出します。

演算には乗除算器を使用します。乗除算器の動作については4.8項を参照してください。

A/D変換結果の平均値から乗除算器を使用して摂氏温度を算出し、温度の値と符号をそれぞれRAMに保存します。温度の計算については4.7項を参照してください。

演算には乗除算器を使用します。乗除算器の動作については4.8項を参照してください。

- (a) 引き算では+/-の変化を考慮して場合分けをし、計算します。

LEDの輝度制御のため、A/D変換結果の平均値をPWM出力のスレーブ・チャンネルのタイマ・データ・レジスタ（TDR07）に設定し、デューティ比を更新します。PWM出力については4.9項を参照してください。

測定できる温度の範囲は - 34.08 から99.95 です。

```

HALT                ; wait 500ms & cut electric power consumption

BTCLR      TMIF00, $LMAIN010      ; 500ms elapsed ? , Yes
BR   LMAIN500                    ; No

LMAIN010:
;-----;
; get A/D conversion result      ;
;-----;
(a) SET1 ADCS          ; A/D converter enable
CLR1 ADIF          ; clear A/D interrupt request flag
CLR1 ADMK          ; enable A/D interrupt
HALT              ; set HALT mode and wait conversion finished
SET1 ADMK          ; disable A/D interrupt
CLR1 ADIF          ; clear A/D interrupt request flag

; get A/D conversion result
(b) MOV B,          RSAVE          ; get A/D result save position
SHL B,          1          ; adjust save position
MOVW AX,        ADCR          ; get A/D result
MOVW RADBUF[B],AX

(c) BF  FFIRSTAD,$LMAIN100      ; first A/D result
CLR1 FFIRSTAD
; initialize A/D result buffer
MOV B,          #(CGETAD * 2)    ; initialize A/D result buffer
LMAIN050:
DEC B          ; initialize A/D result buffer
DEC B
CMP0 B          ; initialization finished ?
BZ  $LMAIN100  ; Yes
MOVW RADBUF[B],AX          ; set A/D result as initial data
BR   LMAIN050

LMAIN100:
(d) INC RSAVE          ; A/D result save position update
CMP RSAVE, #CGETAD      ; save position is over buffer ?
SKC          ; No
CLRB RSAVE          ; Yes (set top of buffer)

;-----;
; calculate A/D result average  ;
;-----;
CLRB B          ; clear position of A/D result getting
CLRW AX
MOVW RWORK, AX          ; clear work area(low 16bit)
LMAIN200:
MOVW AX,        RADBUF[B]
ADDW AX,        RWORK
MOVW RWORK, AX
INC B          ; update position of A/D result getting
INC B
MOV A,          B
SHR A,          1
CMP A,          #CGETAD      ; finished calculation of A/D result sum ?
BC  $LMAIN200          ; No
    
```

リセット・スタート後、1回目A/D変換結果はバッファの全要素に初期値として保存します。これにより、1回目から9回目のA/D変換の後の平均の計算で、A/D変換結果とかけ離れた値が算出されるのを防ぎます。この処理はリセット・スタート後1回目のA/D変換結果を読み出したときのみ動作します。


```

SET1 DIVMODE           ; set division mode
MOVW AX,      RWORK
MOVW MDAL,    AX           ; set dividend(low 16bit)
MOVW MDAH,    #0           ; set dividend(high 16bit)
MOVW MDBL,    #CGETAD     ; set divisor(low 16bit)
MOVW MDBH,    #0           ; set divisor(high 16bit)
SET1 DIVST           ; division start
MOVW HL,      #LOWW MDUC
BT  [HL].0, $$           ; wait for division to be finished
MOVW AX,      MDAL
MOVW RADA VE, AX           ; set average of A/D result (12bit)

;-----;
; calculate temperature(degrees celsius) * 100 ;
; ;
;          CCALC1 - CCALC2 * RADA VE ;
; RTEMPERATURE = ----- ;
;          CCALC3 ;
;-----;
;----- RWORK = CCALC2 * RADA VE
CLR1 DIVMODE           ; set multiplication mode
MOVW MDAL,    #CCALC2     ; set multiplicand
MOVW AX,      RADA VE
MOVW MDAH,    AX           ; set multiplier (multiplication start)
NOP                ; wait for multiplication to be finished(more than 1clk)
MOVW AX,      MDBL
MOVW RWORK,AX           ; get product(low 16bit)
MOVW AX,      MDBH
MOVW (RWORK + 2),AX     ; get product(high 16bit)

(a) ;----- RWORK = CCALC1 - RWORK
MOVW AX,      #HIGHW CCALC1
CMPW AX,      (RWORK + 2)
BC  $LMAIN350           ; RWORK > CCALC1
BNZ $LMAIN300           ; RWORK < CCALC1
MOVW AX,      #LOWW CCALC1
CMPW AX,      RWORK
BC  $LMAIN350           ; RWORK > CCALC1

LMAIN300:           ; RWORK <= CCALC1
CLR1 FMINUS           ; set sign plus
MOVW AX,      #LOWW CCALC1
SUBW AX,      RWORK           ; convert temperature (low 16bit)
MOVW RWORK,   AX
MOVW AX,      #HIGHW CCALC1
SKNC                ; carry from calculation of low 16 bit ? , No
DECW AX                ; Yes . dec carry from high 16bit
SUBW AX,      (RWORK + 2)     ; convert temperature (high 16bit)
MOVW (RWORK + 2),AX
BR  LMAIN400

```

```

(a) LMAIN350:      ; RWORK > CCALC1
      SET1 FMINUS      ; set sign minus
      MOVW AX,        RWORK
      SUBW AX,        #LOWW CCALC1  ; convert temperature (low 16bit)
      MOVW RWORK,    AX
      MOVW AX,        (RWORK + 2)
      SKNC              ; carry from calculation of low 16 bit ? , No
      DECW AX          ; Yes . dec carry from high 16bit
      SUBW AX,        #HIGHW CCALC1 ; convert temperature (high 16bit)
      MOVW (RWORK + 2),AX

LMAIN400:
      ;----- RTEMPERATURE = RWORK / CCALC3
      SET1 DIVMODE      ; set division mode
      MOVW AX,        RWORK
      MOVW MDAL,      AX          ; set dividend(low 16bit)
      MOVW AX,        (RWORK + 2)
      MOVW MDAH,      AX          ; set dividend(high 16bit)
      MOVW MDBL,      #CCALC3    ; set divisor(low 16bit)
      MOVW MDBH,      #0         ; set divisor(high 16bit)
      SET1 DIVST       ; division start
      MOVW HL,        #LOWW MDUC
      BT [HL].0, $$      ; wait for division to be finished
      MOVW AX,        MDAL
      MOVW RTEMPERATURE,AX      ; save temperature * 100
      ; adjust [-0.00 degrees] to [0.00 degrees]
      CMPW AX,        #0         ; 0.00 degrees ?
      SKNZ              ; No
      CLR1 FMINUS      ; set sign plus(0.00 degrees)

      ;*****;
      ; The calculation of the temperature was completed. ;
      ; Please refer to ucSign(+/-) & ushTemperature(temperature * 100). ;
      ;*****;

      ;-----;
      ; PWM output update ;
      ;-----;

      MOVW AX,        RDAVE      ; get average
      MOVW TDR07,    AX          ; PWM output update

LMAIN500:

```

C言語でも、アセンブリ言語と同様な動作を行います。

C言語版のサンプル・プログラムで乗除算器を使用する場合、PM+のオプションで設定を行いますので、ソース・プログラム (main.c) には記述しません。PM+のコンパイラオプションの設定からスタートアップ・ルーチンのタブを選択し、「乗除算器を使用する」にチェックを入れることで、乗除算器を使用する実行ファイルが生成されます。

```
HALT();      /* wait 500ms & cut electric power consumption */

if(TMIF00){
    TMIF00 = 0;

    /*-----*/
    /* get A/D conversion result          */
    /*-----*/
    ADCS = 1; /* A/D converter enable */
    ADIF = 0; /* clear A/D interrupt request flag */
    ADMK = 0; /* enable A/D interrupt */
    HALT(); /* set HALT mode and wait conversion finished */
    ADMK = 1; /* disable A/D interrupt */
    ADIF = 0; /* clear A/D interrupt request flag */

    ushADBuffer[ucSave] = ADCR; /* get A/D result */

    if(bFirstAd){ /* if first A/D conversion */
        bFirstAd = 0; /* clear A/D conversion flag */
        for(loop = 1; loop < CGETAD; loop++){
            ushADBuffer[loop] = ushADBuffer[0]; /* initialize buffer */
        }
    }

    ucSave = ((ucSave + 1) % CGETAD); /* A/D result save position update */

    /*-----*/
    /* calculate A/D result average          */
    /*-----*/
    for(loop = 0, work = 0; loop < CGETAD; loop++){
        work += ushADBuffer[loop];
    }
    ushADAverage = (unsigned short)(work / CGETAD); /* average of A/D result */
}
```

```
/*-----*/
/* calculate temperature(degrees celsius) * 100 */
/*
/*          CCALC1 - CCALC2 * ushADAverage */
/* ushTemperature = ----- */
/*          CCALC3 */
/*-----*/
work = CCALC2 * (unsigned long int)ushADAverage;
if(work > CCALC1){
    ucSign = CMINUS; /* -nn.nn degrees celsius */
    work = work - CCALC1;
}
else{
    ucSign = CPLUS; /* nnn.nn degrees celsius */
    work = CCALC1 - work;
}
ushTemperature=(unsigned short)(work/CCALC3);/* save temperature * 100 */

/* adjust [-0.00 degrees] to [0.00 degrees] */
if(ushTemperature == 0){
    ucSign = CPLUS; /* set sign plus */
}

/*****
/* The calculation of the temperature was completed. */
/* Please refer to ucSign(+/-) & ushTemperature(temperature * 100). */
*****/

/*-----*/
/* PWM output update */
/*-----*/
TDR07 = ushADAverage;
}
```

4.7 温度計算式

温度計算の式について説明します。温度と温度センサの出力電圧の関係と、A/Dコンバータの入力電圧とA/D変換結果の関係を利用し、A/D変換結果から温度計算の式を導きます。

本アプリケーション例では、温度センサとしてセイコーインスツル株式会社製 S-8120Cを使用します。V_{DD}が5.0Vの場合の温度特性は下記の通りです。

表4 - 5 S-8120Cの温度特性 (V_{DD}= 5.0V)

項目	記号	規格	Min.	Typ.	Max.	単位
出力電圧	V _{OUT}	Ta = - 30	1.931	1.9451	1.971	V
		Ta = 30	1.454	1.474	1.494	
		Ta = 100	0.862	0.882	0.902	
温度感度	V _{SE}	- 30 Ta 100	- 8.40	- 8.20	- 8.00	mV/
動作温度範囲	Topr	-	-40	-	100	

注意 演算には表中の標準値 (Typ.) を使用します。

備考 Ta : 温度

S-8120Cは温度変化に対して、リニアな出力電圧が得られるのが特徴です。よって温度による出力電圧の変化は比例することがわかります。Ta = - 30 のときの出力電圧 (V_{OUT}) と、温度感度 (V_{SE}) より、出力電圧温度変化に対する出力電圧の変化は次式に示す関係になります。

$$V_{OUT} = - 0.0082 \times Ta + 1.72$$

$$Ta = (1.72 - V_{OUT}) \div 0.0082 \dots \dots \dots [式1]$$

アナログ入力端子 (ANI0-ANI10, ANI15) に入力されたアナログ入力電圧と理論上のA/D変換結果 (12ビットA/D変換結果レジスタ (ADCR)) には次式に示す関係があります。

$$ADCR = INT \left(\frac{V_{AIN}}{V_{REFOUT}} \times 4096 \right)$$

$$V_{AIN} = (ADCR \times V_{REFOUT}) \div 4096$$

$$V_{AIN} = ADCR \div 2048 \dots \dots \dots [式2]$$

- INT () : () 内の値の整数部を返す関数
- V_{AIN} : アナログ入力電圧
- V_{REFOUT} : ボルテージ・リファレンス出力電圧 (2.0V)
- ADCR : 12ビットA/D変換結果レジスタ (ADCR) の値

A/D変換の対象は温度センサの出力電圧ですので、 $V_{OUT} = V_{AIN}$ となります。式1と式2より、A/D変換結果から温度を算出する計算式は次式になります。

$$\begin{aligned}
 Ta &= (1.72 - (ADCR \div 2048)) \div 0.0082 \\
 &= ((3522.56 - ADCR) \div 2048) \div 0.0082 \\
 &= (3522.56 - ADCR) \div 16.7936 \\
 &= ((3522.56 - ADCR) \times 10^4) \div (16.7936 \times 10^4) \\
 &= (35225600 - 10000 \times ADCR) \div 167936 \\
 &= (2201600 \times 16 - 625 \times 16 \times ADCR) \times 4 \div (10496 \times 16) \\
 Ta &= (2201600 - 625 \times ADCR) \div 10496 \dots\dots\dots [式3]
 \end{aligned}$$

本サンプル・プログラムでは、温度は少数第2位まで算出し、100倍した値をRAMに保存します。100倍した温度を計算するため、式3の両辺を100倍した次式をプログラムで使用します。

$$\begin{aligned}
 Ta \times 100 &= ((2201600 - 625 \times ADCR) \times 100) \div 10496 \\
 &= ((2201600 \times 25 - 625 \times 25 \times ADCR) \times 4) \div (2624 \times 4) \\
 Ta \times 100 &= (55040000 - 15625 \times ADCR) \div 2624
 \end{aligned}$$

温度計算の式で利用する定数は、プログラムの中で以下のようにシンボル定義をして使用しています。

アセンブリ言語版

```

CCALC1 EQU 55040000 ; CCALC1 - CCALC2 * AD
CCALC2 EQU 15625 ; C * 100 = -----
CCALC3 EQU 2624 ; CCALC3
    
```

C言語版

```

#define CCALC1 55040000 /* CCALC1 - CCALC2 * AD */
#define CCALC2 15625 /* C*100=----- */
#define CCALC3 2624 /* CCALC3 */
    
```

4.8 乗除算器の使用方法

乗除算器の使用方法について説明します。

乗除算器には、次のような機能があります。

- ・ 16ビット×16ビット = 32ビット (乗算)
- ・ 32ビット÷32ビット = 32ビット 剰余32ビット (除算)

(1) 乗算器の使用方法

乗算時のレジスタ構成は以下のようになっています。

$$\begin{array}{ccc}
 \text{< 乗数A >} & \text{< 乗数B >} & \text{< 積 >} \\
 \text{MDAL (ビット15-0)} \times \text{MDAH (ビット15-0)} = & \text{[MDBH (ビット15-0), MDBL (ビット15-0)]}
 \end{array}$$

以下に乗算の使用方法について説明します。

・初期設定

乗除算コントロール・レジスタ (MDUC) のビット7 (DIVMODE) を0にする。

乗除算データ・レジスタA (L) (MDAL) に被乗数をセット

乗除算データ・レジスタA (H) (MDAH) に乗数をセット

(, のセットの順はどちらが先でも問題ありません。MDAH, MDALに乗数, 被乗数をセットすると自動的に乗算演算を開始します。)

・演算処理中

1クロック以上ウエイトします。演算は1クロックで終了します。

・演算終了

乗除算データ・レジスタB (L) (MDBL) から積 (下位16ビット) を読み出します。

乗除算データ・レジスタB (H) (MDBH) から積 (上位16ビット) を読み出します。

(, の読み出しの順はどちらが先でも問題ありません。)

・次回演算

次に乗算を行う場合は、乗算動作の「初期設定」から行ってください。

次に除算を行う場合は、(2) 除算器の使用の「初期設定」から行ってください。

(2) 除算器の使用方法

除算時のレジスタ構成は以下のようになっています。

$$\begin{array}{r}
 \begin{array}{cc}
 \text{< 被除数 >} & \text{< 除数 >} \\
 [\text{MDAH (ビット15-0)}, \text{MDAL (ビット15-0)}] & \div [\text{MDBH (ビット15-0)}, \text{MDBL (ビット15-0)}] = \\
 \text{< 商 >} & \text{< 剰余 >} \\
 [\text{MDAH (ビット15-0)}, \text{MDAL (ビット15-0)}] \dots [\text{MDCH (ビット15-0)}, \text{MDCL (ビット15-0)}]
 \end{array}
 \end{array}$$

以下に除算器の使用方法について説明します。

・初期設定

- 乗除算コントロール・レジスタ (MDUC) のビット7 (DIVMODE) に1をセットする。
 - 乗除算データ・レジスタA (H) (MDAH) に被除数 (上位16ビット) をセットする。
 - 乗除算データ・レジスタA (L) (MDAL) に被除数 (下位16ビット) をセットする。
 - 乗除算データ・レジスタB (H) (MDBH) に除数 (上位16ビット) をセットする。
 - 乗除算データ・レジスタB (L) (MDBL) に除数 (下位16ビット) をセットする。
 - MDUCのビット0 (DIVST) に1をセットする。
- (~ の順はどれからセットしても問題ありません。)

・演算処理中

- 次のいずれかの処理が完了すれば演算が終了します。
 - ・ 16クロック以上ウエイト (16クロックで演算は終了します。)
 - ・ DIVSTがクリアされたことを確認
 - ・ 除算完了割り込み (INTMD) 発生
- (演算処理中のMDBL, MDBH, MDCH, MDCLのリード値は保証しません。)

・演算終了

- DIVSTがクリア (0) され、割り込み要求信号 (INTMD) が発生します (演算終了)。
 - MDALから商 (下位16ビット) を読み出します。
 - MDAHから商 (上位16ビット) を読み出します。
 - 乗除算データ・レジスタC (L) (MDCL) から剰余 (下位16ビット) を読み出します。
 - 乗除算データ・レジスタC (H) (MDCH) から剰余 (上位16ビット) を読み出します。
- (~ の順はどれから読み出しても問題ありません。)

・次回演算

- 次に乗算を行う場合は、(1) 乗算器の使用の「初期設定」から行ってください。
- 次に除算を行う場合は、除算動作の「初期設定」から行ってください。

4.9 PWM機能としての動作

PWM機能としての動作について説明します。

PWM機能では、2チャンネルをセットで使用し、任意の周期およびデューティのパルスを生成することができます。このプログラムではマスタ・チャンネルとしてチャンネル6を、スレーブ・チャンネルとしてチャンネル7を使用します。出力パルスの周期、デューティは次の式で求めることができます。

<p>パルス周期 = { TDR06 (マスタ) の設定値 + 1 } × カウント・クロック周期</p> <p>デューティ [%] = { TDR07 (スレーブ) の設定値 } / { TDR06 (マスタ) の設定値 + 1 } × 100</p> <p>0%出力 : TDR07 (スレーブ) の設定値 = 0000H</p> <p>100%出力 : TDR07 (スレーブ) の設定値 { TDR06 (マスタ) の設定値 + 1 }</p>
--

備考 TDR07 (スレーブ) の設定値 > { TDR06 (マスタ) の設定値 + 1 } の場合は、デューティ値が100%を越えますが、集約して100%出力となります。

マスタ・チャンネルは、インターバル・タイマ・モードで動作して、周期をカウントします。チャンネル・スタート・トリガ (TS06) に1を設定すると、INTTM06を出力します。TCR06はロードしたTDR06の値からカウント・クロックに合わせてダウン・カウントを行います。TCR06 = 0000Hとなったら、INTTM06を出力し、再びTCR06はTDR06の値をロードして、以降、同様の動作を続けます。

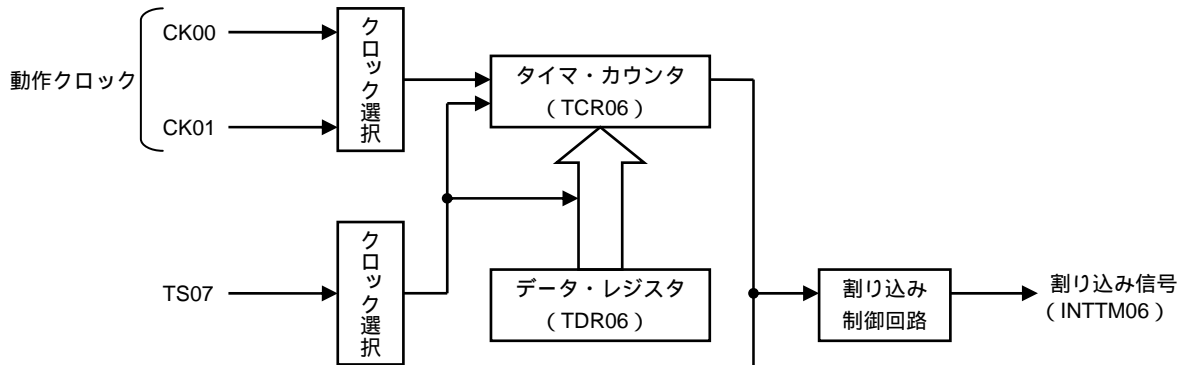
スレーブ・チャンネルは、ワンカウント・モードで動作して、デューティをカウントし、TO07端子よりPWM波形を出力します。スレーブ・チャンネルのTCR07は、マスタ・チャンネルのINTTM06をスタート・トリガとして、TDR07の値をロードし、ダウン・カウントを行います。TCR07 = 0000Hとなったら、INTTM07を出力し、次のスタート・トリガ (マスタ・チャンネルのINTTM06) が入力されるまでカウントを停止します。TO07の出力レベルは、マスタ・チャンネルのINTTM06発生から1カウント・クロック経過後にアクティブ・レベルとなり、TCR07 = 0000Hとなったらインアクティブ・レベルとなります。

注意 マスタ・チャンネルのTDR06とスレーブ・チャンネルのTDR07を両方とも書き換える場合、2回のライト・アクセスが必要となります。TCR06, TCR07にTDR06, TDR07の値がロードされるのは、マスタ・チャンネルのINTTM06発生時となります。そのため、書き換えがマスタ・チャンネルのINTTM06発生前と発生後に分かれて行われるとTO07端子は期待通りの波形を出力できません。したがって、マスタのTDR06とスレーブのTDR07を双方とも書き換える場合は、必ずマスタ・チャンネルのINTTM06発生直後に両方のレジスタを書き換えてください。

図4 - 19 PWM機能としての動作のブロック図

マスタ・チャンネル

(インターバル・タイマ・モード)



スレーブ・チャンネル

(ワンカウント・モード)

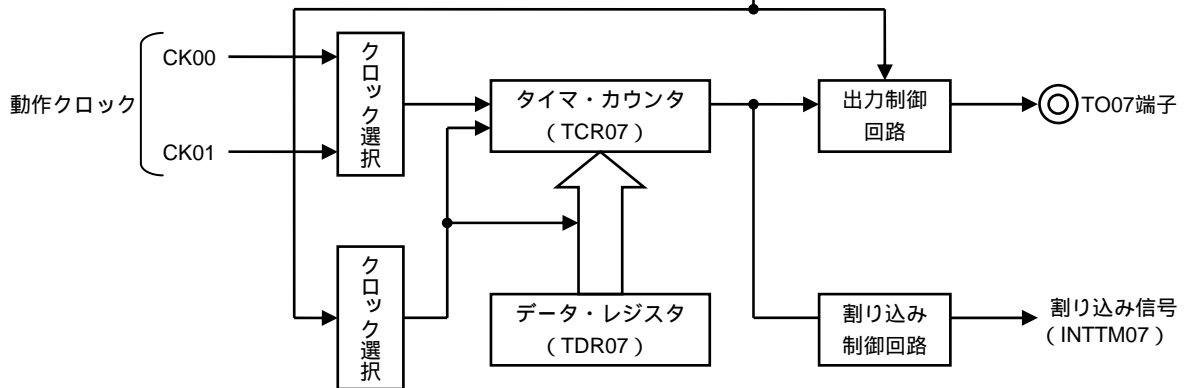
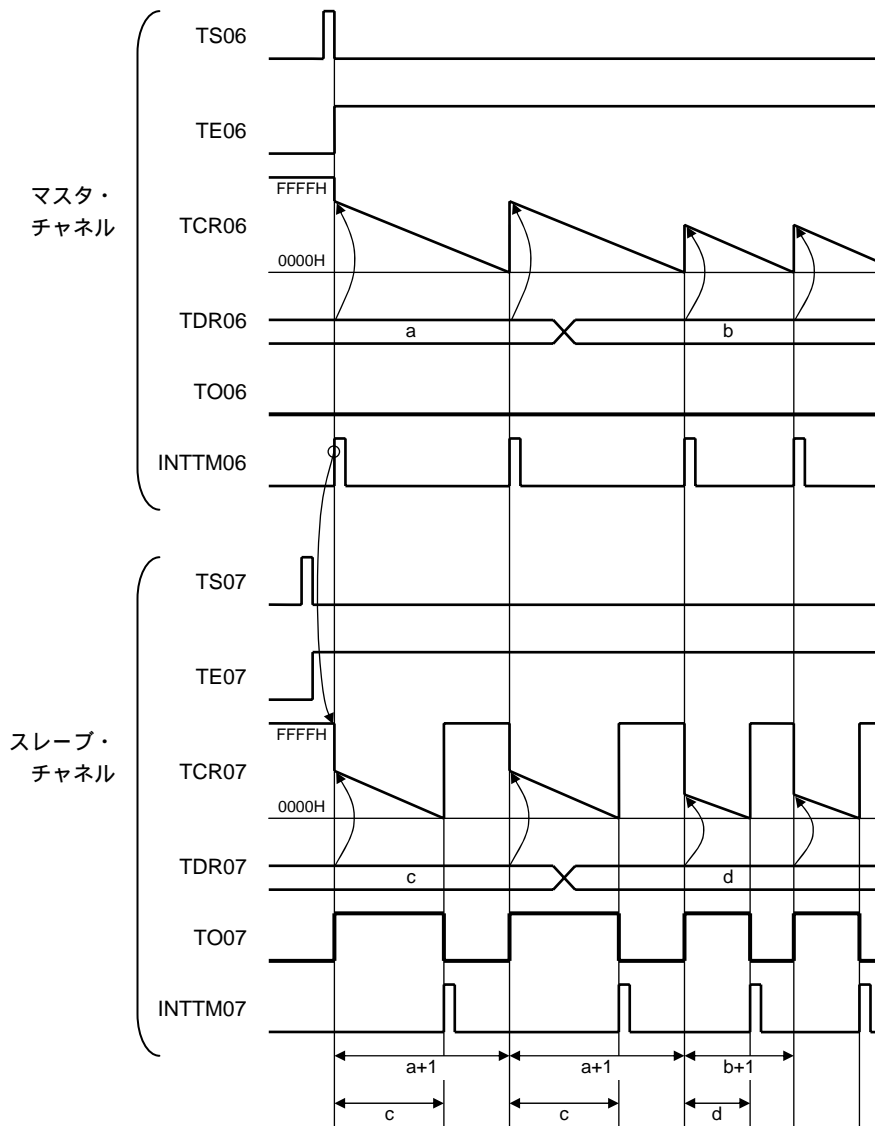


図4 - 20 PWM機能としての動作の基本タイミング例



4.10 PWM出力によるLEDの輝度制御

PWM出力によるLEDの輝度制御について説明します。

LEDの輝度は、PWM出力のデューティ比によって変化します。

デューティ比をA/D変換結果の値で変化させるため、A/D変換結果が最大（FFFH）のときにデューティ比が100%となるように、TDR06（マスタ・チャンネル）にFFFH - 1を設定します。TDR07（スレーブ・チャンネル）は、A/D変換結果を設定します。

よって、デューティ比は次の式に示す関係になります。

$$\text{デューティ}[\%] = \text{A/D変換結果} \div \text{FFFH} \times 100$$

備考 A/D変換結果には、10回分のA/D変換で取得したA/D変換結果レジスタの値の平均値を設定します。

表4 - 6 A/D変換結果、温度、およびデューティ比の関係の例

A/D変換結果	温度	デューティ比	LEDの状態
734H	99.95	45.02%	
800H	87.80	50.01%	
900H	72.56	56.26%	
A00H	57.31	62.52%	
B00H	42.07	68.77%	
C00H	26.82	75.02%	
D00H	11.58	81.27%	
E00H	- 3.65	87.52%	
F00H	- 18.90	93.77%	
FFFH	- 34.08	100.00%	

温度が上がるとデューティ比は小さくなり、TO07端子からロウ・レベルを出力する時間が長くなります。PWM出力で制御するLEDはロウ・アクティブなので、温度が高いほどLEDは明るくなります。

第5章 関連資料

資料名		和文 / 英文
78K0R/Lx3 ユーザーズ・マニュアル		PDF
78K0Rシリーズ 命令編 ユーザーズ・マニュアル		PDF
RA78K0R アセンブラ・パッケージ ユーザーズ・マニュアル	言語編	PDF
	操作編	PDF
CC78K0R Cコンパイラ ユーザーズ・マニュアル	言語編	PDF
	操作編	PDF
PM+ プロジェクト・マネージャ ユーザーズ・マニュアル		PDF

付録A プログラム・リスト

プログラム・リスト例として、ソース・プログラムを次に示します。

```
main.asm (アセンブリ言語版)
;
; Copyright (C) NEC Electronics Corporation 2006
; NEC ELECTRONICS CONFIDENTIAL AND PROPRIETARY
; All rights reserved by NEC Electronics Corporation.
; This program must be used solely for the purpose for which
; it was furnished by NEC Electronics Corporation. No part of this
; program may be reproduced or disclosed to others, in any
; form, without the prior written permission of NEC Electronics
; Corporation. Use of copyright notice dose not evidence
; publication of the program.
;
;-----;
; Extern function ;
;-----;
;-----;
; Extern variables/constants ;
;-----;
;-----;
; Vector table initialize ;
;-----;
TVCT1 CSEG AT 000000H
      DW IRESET ;(00H) RESET, POC, LVI, WDT, TRAP
TVCT2 CSEG AT 000004H
      DW IRESET ;(04H) INTWDTI
      DW IRESET ;(06H) INTLVI
      DW IRESET ;(08H) INTP0
      DW IRESET ;(0AH) INTP1
      DW IRESET ;(0CH) INTP2
      DW IRESET ;(0EH) INTP3E
      DW IRESET ;(10H) INTP4
      DW IRESET ;(12H) INTP5
      DW IRESET ;(14H) INTST3
      DW IRESET ;(16H) INTSR3
      DW IRESET ;(18H) INTSRE3
```

```

DW      IRESET      ;(1AH) INTDMA0
DW      IRESET      ;(1CH) INTDMA1
DW      IRESET      ;(1EH) INTST0, INTCSI00
DW      IRESET      ;(20H) INTSR0, INTCSI01
DW      IRESET      ;(22H) INTSRE0
DW      IRESET      ;(24H) INTST1, INTCSI10, INTIIC10
DW      IRESET      ;(26H) INTSR1
DW      IRESET      ;(28H) INTSRE1
DW      IRESET      ;(2AH) INTIICA
DW      IRESET      ;(2CH) INTTM00
DW      IRESET      ;(2EH) INTTM01
DW      IRESET      ;(30H) INTTM02
DW      IRESET      ;(32H) INTTM03
DW      IRESET      ;(34H) INTAD
DW      IRESET      ;(36H) INTRTC
DW      IRESET      ;(38H) INTRTCI
DW      IRESET      ;(3AH) INTKR
DW      IRESET      ;(3CH) INTST2, INTCSI20, INTIIC20
DW      IRESET      ;(3EH) INSR2
DW      IRESET      ;(40H) INTSRE2
DW      IRESET      ;(42H) INTTM04
DW      IRESET      ;(44H) INTTM05
DW      IRESET      ;(46H) INTTM06
DW      IRESET      ;(48H) INTTM07
DW      IRESET      ;(4AH) INTP6
DW      IRESET      ;(4CH) INTP7
DW      IRESET      ;(4EH) INTP8
DW      IRESET      ;(50H) INTP9
DW      IRESET      ;(52H) INTP10
DW      IRESET      ;(54H) INTP11
DW      IRESET      ;(56H) INTTM10
DW      IRESET      ;(58H) INTTM11
DW      IRESET      ;(5AH) INTTM12
DW      IRESET      ;(5CH) INTTM13
DW      IRESET      ;(5EH) INTMD
TVCT3   CSEG      AT      00007EH
        DW      IRESET      ;(7EH) BRK

```

-----;

; Stack area definition ;

-----;

DSTK DSEG IHRAM ;Stack Area address

STACKEND:

```

DS      60H

STACKTOP:

;-----;
; Local constants                                     ;
;-----;
; *****.
; * data for calculation to convert                *;
; * [12-bit A/D conversion result]                *;
; * into [temperature(degrees celsius) * 100].    *;
; *****.
; Legend symbol
; AD: 12-bit A/D conversion result
; VAIN: Analog input of the A/D converter
;      = Output voltage of temperature sensor
; C:   Temperature (degrees celsius)
;
; -----
; Features of temperature sensor (S-8120C)
; * Linear output voltage for temperature changes.
; * Temperature sensitivity : -8.2[mV/C] (Typ.)
; * Output voltage : Ta = 30[°C] -> 1.474[V] (Typ.)
; -----
;
; -> VAIN = -0.0082 * C + 1.72
;
;      (1.72 - VAIN) * 10000      8600 - (5000 * VAIN)
; C = ----- = -----
;           82                    41
;
; -----
; Setting of A/D converter
; * AVREF : 2.0[V]
; * 12-bit resolution (result : 000H-FFFH)
; -----
;
; -> AD = INT((VAIN/AVREF)*4096)
; VAIN = (AD / 4096) * AVREF
;      = AD / 2048
;
;      8600 - 5000 * (AD / 2048)      2201600 - 625 * AD
; C = ----- = -----
;           41                    10496

```



```

;
;          2201600 - 625 * AD          55040000 - 15625 * AD
;  C * 100 = ----- * 100 = -----
;          10496                      2624
;
;
CCALC1 EQU 55040000 ;          CCALC1 - CCALC2 * AD
CCALC2 EQU 15625    ;  C * 100 = -----
CCALC3 EQU 2624    ;          CCALC3

;-----;
; Global variables ;
;-----;
;-----;
; Local variables ;
;-----;
DMAIN DSEG SADDR
      RSAVE: DS 1 ; Save position of A/D result
          CGETAD EQU 10 ; get A/D result number of times

DPMAIN DSEG SADDRP
      RTEMPERATURE: DS 2 ; Temperature * 100 @ HEX
      RADBUF: DS (CGETAD*2) ; A/D result buffer
      RADAVE: DS 2 ; Average of A/D result
      RWORK: DS 4 ; Calculation area

DBMAIN BSEG UNIT
      FMINUS DBIT ; sign of temperature(0:+,1:-)
      FFIRSTAD DBIT ; first A/D conversion flag

;-----;
; Code ;
;-----;
XMAIN CSEG UNIT
;-----;
; Hardware initialization ;
;-----;
      PUBLIC IRESET
IRESET:
;-----;
; Disable all interrupts ;
;-----;
DI

```

```

;-----;
;           Set register bank           ;
;-----;
SEL      R0           ; use register bank 0

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

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

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

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

```

```

;|||
;||+----- : Be sure to set 0
;||
;+----- : [1]EXCLK OSCSEL High-speed system clock pin operation mode
;           [2]X1/P121 pin
;           [3]X2/EXCLK/P122 pin
; : 0 0 : [1]Input port mode
;         [2][3]Input port
;
; : 0 1 : [1]X1 oscillation mode
;         [2][3]Crystal/ceramic resonator connection
;
; : 1 0 : [1]Input port mode
;         [2][3]Input port
;
; : 1 1 : [1]External clock input mode
;         [2]Input port
;         [3]External clock input

CLR1    MSTOP                ; X1 oscillator operating

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

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

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

```

```

NOP
NOP

MOV    CKC,    #00010000B    ; System Clock Control Register
        ;|+|+++++--- : Selection of CPU/peripheral hardware clock (fCLK)
        ;| | : 0 0 x 0 0 0 : fIH
        ;| | : 0 0 x 0 0 1 : fIH/2 (default)
        ;| | : 0 0 x 0 1 0 : fIH/2^2
        ;| | : 0 0 x 0 1 1 : fIH/2^3
        ;| | : 0 0 x 1 0 0 : fIH/2^4
        ;| | : 0 0 x 1 0 1 : fIH/2^5
        ;| | : 0 1 x 0 0 0 : fMX
        ;| | : 0 1 x 0 0 1 : fMX/2
        ;| | : 0 1 x 0 1 0 : fMX/2^2
        ;| | : 0 1 x 0 1 1 : fMX/2^3
        ;| | : 0 1 x 1 0 0 : fMX/2^4
        ;| | : 0 1 x 1 0 1 : fMX/2^5
        ;| | : 1 x 0 x x x : fSUB
        ;| | : 1 x 1 x x x : fSUB/2
        ;| | : Other than above : Setting prohibited
        ;| | x = don't care
        ;| |
        ;| +----- : Status of Main system clock (fMAIN)
        ;| : 0 : Internal high-speed oscillation clock (fIH)
        ;| : 1 : High-speed system clock (fMX)
        ;|
        ;+----- : Status of CPU/peripheral hardware clock (fCLK)
        ; : 0 : Main system clock (fMAIN)
        ; : 1 : Subsystem clock (fSUB)

HRST100.; CPU is operating on a High-speed system clock ?
BT      CLS,    $HRST100 ; No
BF      MCS,    $HRST100 ; No

SET1    HI0STOP    ; internal high-speed oscillation stopped

MOV     OSTs,    #00000111B    ; Oscillation Stabilization Time Select Register
        ;|||+|+++++--- : Oscillation stabilization time selection
        ;||| : 0 0 0 : 2^8/fX
        ;||| : 0 0 1 : 2^9/fX
        ;||| : 0 1 0 : 2^10/fX
        ;||| : 0 1 1 : 2^11/fX
        ;||| : 1 0 0 : 2^13/fX
    
```

```

;| | | | : 1 0 1 : 215/fX
;| | | | : 1 1 0 : 217/fX
;| | | | : 1 1 1 : 218/fX
;| | | |
;++++----- : Be sure to set 0000

MOV     PER0,    #00100001B      ; Peripheral Enable Register 0
;| | | | | +--- : Control of timer array unit 0 input clock
;| | | | | : 0 : Stops input clock supply.
;| | | | |      * SFR used by timer array unit 0 cannot be written.
;| | | | |      * Timer array unit 0 is in the reset status.
;| | | | | : 1 : Supplies input clock.
;| | | | |      * SFR used by timer array unit 0 can be read and written.
;| | | | |
;| | | | | +--- : Control of timer array unit 1 input clock
;| | | | | : 0 : Stops input clock supply.
;| | | | |      * SFR used by timer array unit 1 cannot be written.
;| | | | |      * Timer array unit 1 is in the reset status.
;| | | | | : 1 : Supplies input clock.
;| | | | |      * SFR used by timer array unit 1 can be read and written.
;| | | | |
;| | | | +----- : Control of serial array unit 0 input clock
;| | | | : 0 : Stops input clock supply.
;| | | |      * SFR used by the serial array unit 0 cannot be written.
;| | | |      * The serial array unit 0 is in the reset status.
;| | | | : 1 : Supplies input clock.
;| | | |      * SFR used by the serial array unit 0 can be read and written.
;| | | |
;| | | | +----- : Control of serial array unit 1 input clock
;| | | | : 0 : Stops input clock supply.
;| | | |      * SFR used by the serial array unit 1 cannot be written.
;| | | |      * The serial array unit 1 is in the reset status.
;| | | | : 1 : Supplies input clock.
;| | | |      * SFR used by the serial array unit 1 can be read and written.
;| | | |
;| | | +----- : Control of serial interface IICA input clock
;| | | : 0 : Stops input clock supply.
;| | |      * SFR used by the serial interface IICA cannot be written.
;| | |      * The serial interface IICA is in the reset status.
;| | | : 1 : Supplies input clock.
;| | |      * SFR used by the serial interface IICA can be read and written.
;| | |
;| | +----- : Control of A/D converter, operational amplifier, and voltage

```

reference input clock

;|| : 0 : Stops input clock supply.

;|| * SFR used by the A/D converter, operational amplifier, and voltage

reference cannot be written.

;|| * The A/D converter, operational amplifier, and voltage reference

is in the reset status.

;|| : 1 : Supplies input clock.

;|| * SFR used by the A/D converter, operational amplifier, and voltage

reference can be read and written.

;||

;|+----- : Control of D/A converter input clock

;| : 0 : Stops input clock supply.

;| * SFR used by D/A converter cannot be written.

;| * The D/A converter is in the reset status.

;| : 1 : Supplies input clock.

;| * SFR used by the D/A converter can be read and written.

;|

;+----- : Control of real-time counter (RTC) input clock

; : 0 : Stops input clock supply.

; * SFR used by the real-time counter (RTC) cannot be written.

; * The real-time counter (RTC) is in the reset status.

; : 1 : Supplies input clock.

; * SFR used by the real-time counter (RTC) can be read and written.

-----;

; Initialization of timer ;

-----;

CALL !!SINITTAU0

-----;

; Initialization of voltage reference ;

-----;

CALL !!SINITVR

-----;

; Initialization of A/D converter ;

-----;

CALL !!SINITADC

-----;

; Initialization of variable ;

-----;

MOVW RTEMPERATURE,#0 ; initialize temperature value

```

CLR1  FMINUS          ; initialize sign of temperature
CLRB  RSAVE          ; initialize save position of A/D result
MOVW  RADAVE, #0     ; initialize average
SET1  FFIRSTAD ; set A/D conversion flag

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

MMAIN:
HALT                                     ; wait 500ms & cut electric power consumption

BTCLR  TMIF00, $LMAIN010 ; 500ms elapsed ? , Yes
BR     LMAIN500          ; No

LMAIN010:
;-----;
; get A/D conversion result          ;
;-----;
SET1  ADCS           ; A/D converter enable
CLR1  ADIF           ; clear A/D interrupt request flag
CLR1  ADMK           ; enable A/D interrupt
HALT                                     ; set HALT mode and wait conversion finished
SET1  ADMK           ; disable A/D interrupt
CLR1  ADIF           ; clear A/D interrupt request flag

; get A/D conversion result
MOV   B,      RSAVE          ; get A/D result save position
SHL   B,      1             ; adjust save position
MOVW  AX,     ADCR           ; get A/D result
MOVW  RADBUF[B],AX

BF     FFIRSTAD,$LMAIN100 ; first A/D conversion ? , No
CLR1  FFIRSTAD          ; clear A/D conversion flag
; initialize A/D result buffer
MOV   B,      #(CGETAD * 2) ; initialize position set

LMAIN050:
DEC   B           ; initialize position update
DEC   B
CMP0  B           ; initialization finished ?
BZ    $LMAIN100   ; Yes
MOVW  RADBUF[B],AX ; set A/D result as initial data

```

```

BR      LMAIN050

LMAIN100:
INC     RSAVE                ; A/D result save position update
CMP     RSAVE, #CGETAD       ; save position is over buffer ?
SKC                    ; No
CLRB    RSAVE                ; Yes (set top of buffer)

;-----;
; calculate A/D result average ;
;-----;

CLRB    B                    ; clear position of A/D result getting
CLRW    AX
MOVW    RWORK, AX           ; clear work area(low 16bit)

LMAIN200:
MOVW    AX, RADBUF[B]
ADDW    AX, RWORK
MOVW    RWORK, AX
INC     B                    ; update position of A/D result getting
INC     B
MOV     A, B
SHR     A, 1
CMP     A, #CGETAD         ; finished calculation of A/D result sum ?
BC     $LMAIN200           ; No

SET1    DIVMODE             ; set division mode
MOVW    AX, RWORK
MOVW    MDAL, AX           ; set dividend(low 16bit)
MOVW    MDAH, #0           ; set dividend(high 16bit)
MOVW    MDBL, #CGETAD      ; set divisor(low 16bit)
MOVW    MDBH, #0           ; set divisor(high 16bit)
SET1    DIVST              ; division start
MOVW    HL, #LOWW MDUC
BT     [HL].0, $$          ; wait for division to be finished
MOVW    AX, MDAL
MOVW    RDAVE, AX         ; set average of A/D result (12bit)

;-----;
; calculate temperature(degrees celsius) * 100 ;
; ;
;          CCALC1 - CCALC2 * RDAVE ;
; RTEMPERATURE = ----- ;
;          CCALC3 ;

```



```

;-----;
;----- RWORK = CCALC2 * RADAVE
CLR1    DIVMODE                ; set multiplication mode
MOVW    MDAL,    #CCALC2        ; set multiplicand
MOVW    AX,      RADAVE
MOVW    MDAH,    AX            ; set multiplier (multiplication start)
NOP                                           ; wait for multiplication to be finished(more than 1clk)
MOVW    AX,      MDBL
MOVW    RWORK,AX                ; get product(low 16bit)
MOVW    AX,      MDBH
MOVW    (RWORK + 2),AX          ; get product(high 16bit)

;----- RWORK = CCALC1 - RWORK
MOVW    AX,      #HIGHW CCALC1
CMPW    AX,      (RWORK + 2)
BC      $LMAIN350                ; RWORK > CCALC1
BNZ     $LMAIN300                ; RWORK < CCALC1
MOVW    AX,      #LOWW CCALC1
CMPW    AX,      RWORK
BC      $LMAIN350                ; RWORK > CCALC1

LMAIN300:    ; RWORK <= CCALC1
CLR1    FMINUS                  ; set sign plus
MOVW    AX,      #LOWW CCALC1
SUBW    AX,      RWORK          ; convert temperature (low 16bit)
MOVW    RWORK,   AX
MOVW    AX,      #HIGHW CCALC1
SKNC                                         ; carry from calculation of low 16 bit ? , No
DECW    AX                      ; Yes . dec carry from high 16bit
SUBW    AX,      (RWORK + 2)    ; convert temperature (high 16bit)
MOVW    (RWORK + 2),AX
BR      LMAIN400

LMAIN350:    ; RWORK > CCALC1
SET1    FMINUS                  ; set sign minus
MOVW    AX,      RWORK
SUBW    AX,      #LOWW CCALC1    ; convert temperature (low 16bit)
MOVW    RWORK,   AX
MOVW    AX,      (RWORK + 2)
SKNC                                         ; carry from calculation of low 16 bit ? , No
DECW    AX                      ; Yes . dec carry from high 16bit
SUBW    AX,      #HIGHW CCALC1  ; convert temperature (high 16bit)
MOVW    (RWORK + 2),AX

```

LMAIN400:

```

;----- RTEMPERATURE = RWORK / CCALC3
SET1    DIVMODE                ; set division mode
MOVW    AX,    RWORK
MOVW    MDAL,   AX              ; set dividend(low 16bit)
MOVW    AX,    (RWORK + 2)
MOVW    MDAH,   AX              ; set dividend(high 16bit)
MOVW    MDBL,   #CCALC3         ; set divisor(low 16bit)
MOVW    MDBH,   #0              ; set divisor(high 16bit)
SET1    DIVST                ; division start
MOVW    HL,    #LOWW MDUC
BT      [HL].0,  $$              ; wait for division to be finished
MOVW    AX,    MDAL
MOVW    RTEMPERATURE,AX        ; save temperature * 100
; adjust [-0.00 degrees] to [0.00 degrees]
CMPW    AX,    #0              ; 0.00 degrees ?
SKNZ                                ; No
CLR1    FMINUS                ; set sign plus(0.00 degrees)

;*****
;
; The calculation of the temperature was completed.
; Please refer to ucSign(+/-) & ushTemperature(temperature * 100).
;*****
;
;-----;
; PWM output update
;-----;
MOVW    AX,    RADAVE          ; get average
MOVW    TDR07, AX              ; PWM output update

```

LMAIN500:

```

;*****
;
;*
;* The main processing writes here
;* if there is something
;*
;*****
;
BR      MMMAIN                ; continue main process

```

;-----;

```

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

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

;-----
;   Setting of Port 2
;-----
MOV     P2,     #00000000B      ; Set P20-P27 Output latch to Low
MOV     PM2,    #11111110B      ; Set P20 to output port, P21-P27 to input port
                                           ; P21:Analog input for sensor(ANI1)
                                           ; P20,P22-P27:Unused

;-----
;   Setting of Port 3
;-----
MOV     P3,     #00001000B      ; Set P30-P32,P34 Output latch to Low
                                           ; Set P33 Output latch High
MOV     PM3,    #11100000B      ; Set P30-P34 to output port
                                           ; P33:LED controled by PWM output(T007)
                                           ; P30-P32,P34:Unused

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

```

```

;-----
;   Setting of Port 5
;-----
MOV    P5,    #00000000B    ; Set P50-P57 Output latch to Low
MOV    PM5,   #11110000B    ; Set P50-P57 to output port
                                ; P50-P57:Unused

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

;-----
;   Setting of Port 7
;-----
MOV    P7,    #00000000B    ; Set P70-P77 Output latch to Low
MOV    PM7,   #00000000B    ; Set P70-P77 to output port
                                ; P70-P77:Unused

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

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

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

```

```

;-----
;   Setting of Port 11
;-----
MOV    P11,    #00000000B    ; Set P110-P111 Output latch to Low
MOV    PM11,   #11111100B    ; Set P110-P111 to output port
                                   ; P110-P111:Unused

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

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

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

;-----
;   Setting of Port 15
;-----
MOV    P15,    #00000000B    ; Set P150-P152,P157 Output latch to Low
MOV    PM15,   #11111111B    ; Set P150-P152,P157 to input port
                                   ; P150-P152,P157:Unused

RET

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

```

SINITLVI:

; setting of Low-Voltage Detector

```

SET1    LVIMK                ; disable LVI interrupt
CLR1    LVISEL              ; detects level of VDD
MOV     LVIS, #00001001B    ; Low-Voltage Detection Level Select Register
;||||+++--- : Detection level
;|||| : 0 0 0 0 : VLVI0 (4.22V)
;|||| : 0 0 0 1 : VLVI1 (4.07V)
;|||| : 0 0 1 0 : VLVI2 (3.92V)
;|||| : 0 0 1 1 : VLVI3 (3.76V)
;|||| : 0 1 0 0 : VLVI4 (3.61V)
;|||| : 0 1 0 1 : VLVI5 (3.45V)
;|||| : 0 1 1 0 : VLVI6 (3.30V)
;|||| : 0 1 1 1 : VLVI7 (3.15V)
;|||| : 1 0 0 0 : VLVI8 (2.99V)
;|||| : 1 0 0 1 : VLVI9 (2.84V)
;|||| : 1 0 1 0 : VLVI10 (2.68V)
;|||| : 1 0 1 1 : VLVI11 (2.53V)
;|||| : 1 1 0 0 : VLVI12 (2.38V)
;|||| : 1 1 0 1 : VLVI13 (2.22V)
;|||| : 1 1 1 0 : VLVI14 (2.07V)
;|||| : 1 1 1 1 : VLVI15 (1.91V)
;||||
;++++----- : Be sure to set 0000

```

```

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

```

```

SET1    LVION                ; enables low-voltage detection operation

```

; software to wait for the operation stabilization time (over 10us)

```

MOV     B, #10                ; set counter

```

HRES100:

```

NOP                                ; (1clk)
DEC     B                            ; (1clk)
BNZ     $HRES100                    ; finished waiting ? , No (2clk/4clk)

```

; wait for VDD to become VLVI or more

HRES300:

```

NOP
BT     LVIF, $HRES300 ; VDD < VLVI? , Yes
CLR1    LVION                ; disables low-voltage detection operation

```

RET

```

;-----;
; Module:      SINITTAUO      ;
; Description:  Setting of Timer array unit 0      ;
;   parameter: --      ;
;   return   : --      ;
;-----;
SINITTAUO:
    MOV     TPSOL, #00001000B      ; Timer Clock Select Register 0
           ;|||+--- : Selection of operation clock (CK00)
           ;+++----- : Selection of operation clock (CK01)
           ; : 0 0 0 0 : CK0m = fCLK
           ; : 0 0 0 1 : CK0m = fCLK/2
           ; : 0 0 1 0 : CK0m = fCLK/2^2
           ; : 0 0 1 1 : CK0m = fCLK/2^3
           ; : 0 1 0 0 : CK0m = fCLK/2^4
           ; : 0 1 0 1 : CK0m = fCLK/2^5
           ; : 0 1 1 0 : CK0m = fCLK/2^6
           ; : 0 1 1 1 : CK0m = fCLK/2^7
           ; : 1 0 0 0 : CK0m = fCLK/2^8
           ; : 1 0 0 1 : CK0m = fCLK/2^9
           ; : 1 0 1 0 : CK0m = fCLK/2^10
           ; : 1 0 1 1 : CK0m = fCLK/2^11
           ; : 1 1 0 0 : CK0m = fCLK/2^12
           ; : 1 1 0 1 : CK0m = fCLK/2^13
           ; : 1 1 1 0 : CK0m = fCLK/2^14
           ; : 1 1 1 1 : CK0m = fCLK/2^15
           ; m = 0, 1

;-----;
;   CHO:for timing      ;
;-----;
MOVW     AX, #0000000000000000B ; Timer Mode Register 00
MOVW     TMR00,AX;|||||||+--- : [1]Operation mode of channel 0
           ;||||||| [2]Count operation of TCR
           ;||||||| [3]Independent operation
           ;||||||| [4]Setting of starting counting and interrupt
           ;||||||| : 0 0 0 0 : [1]Interval timer mode
           ;||||||| [2]Counting down
           ;||||||| [3]Possible
           ;||||||| [4]Timer interrupt is not generated
           ;||||||| when counting is started
           ;||||||| (timer output does not change, either).
           ;|||||||

```

```

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

```



```

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

```

```

;|||+----- : Selection of count clock (TCLK) of channel 0
;||| : 0 : Operation clock MCK specified by CKS00 bit
;||| : 1 : Valid edge of input signal input from T100 pin, fSUB/2, fSUB/4, or
INTRTC1
;||| (the timer input used with channel 0 is selected by using T1S0
register).
;|||
;|+----- : Be sure to set 00
;|
;+----- : Selection of operation clock (MCK) of channel 0
; : 0 : Operation clock CK00 set by TPS0 register
; : 1 : Operation clock CK01 set by TPS0 register

; CK00 = fCLK/2^8 = 78.1kHz -> 12.804[us/cik] * 39050[count] = 499.996ms
MOVW TDR00, #(39050 - 1) ; set interval time to about 500 ms

CLR1 TMMK00 ; enable interrupt
SET1 TSOL.0 ; start TAU0 CHO
SET1 TMIF00 ; set interrupt request flag

;-----;
; CH1:for VR settling time ;
;-----;
MOVW AX, #0000000000000000B ; Timer Mode Register 01
MOVW TMR01,AX;|||+---- : [1]Operation mode of channel 1
;||| [2]Count operation of TCR
;||| [3]Independent operation
;||| [4]Setting of starting counting and interrupt
;||| : 0 0 0 0 : [1]Interval timer mode
;||| [2]Counting down
;||| [3]Possible
;||| [4]Timer interrupt is not generated
;||| when counting is started
;||| (timer output does not change, either).
;|||
;||| : 0 0 0 1 : [1]Interval timer mode
;||| [2]Counting down
;||| [3]Possible
;||| [4]Timer interrupt is generated
;||| when counting is started
;||| (timer output also changes).
;|||
;||| : 0 1 0 0 : [1]Capture mode

```

```

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

```

```

;||||| At that time interrupt
;||||| is not generated, either.
;|||||
;||||| : Other than above : Setting prohibited
;|||||
;|||||++----- : Be sure to set 00
;|||||
;|||||++----- : Selection of TI01 pin input signal,
;||||| fSUB/2, fSUB/4, or INTRTC1 valid edge
;||||| (the timer input used with channel 1
;||||| is selected by using TIS0 register).
;||||| : 0 0 : Falling edge
;||||| : 0 1 : Rising edge
;||||| : 1 0 : Both edges (when low-level width is measured)
;||||| Start trigger: Falling edge, Capture trigger: Rising edge
;||||| : 1 1 : Both edges (when high-level width is measured)
;||||| Start trigger: Rising edge, Capture trigger: Falling edge
;|||||
;|||||++----- : Setting of start trigger or capture trigger of channel

```

1

```

;||||| : 0 0 0 : Only software trigger start is valid
;||||| (other trigger sources are unselected).
;||||| : 0 0 1 : Valid edge of TI01 pin input signal, fSUB/2, fSUB/4, or INTRTC1
;||||| is used as both the start trigger and capture trigger.
;||||| : 0 1 0 : Both the edges of TI01 pin input signal, fSUB/2, fSUB/4, or

```

INTRTC1

```

;||||| are used as a start trigger and a capture trigger.
;||||| : 1 0 0 : Interrupt signal of the master channel is used
;||||| (when the channel is used as a slave channel
;||||| with the combination operation function).
;||||| : Other than above : Setting prohibited
;|||||
;|||||++----- : Selection of slave/master of channel 1
;||||| : 0 : Operates as slave channel with combination operation function.
;||||| : 1 : Operates as master channel with combination operation function.
;|||||
;|||||++----- : Selection of count clock (TCLK) of channel 0
;||||| : 0 : Operation clock MCK specified by CKS01 bit
;||||| : 1 : Valid edge of input signal input from TI01 pin, fSUB/2, fSUB/4, or

```

INTRTC1

(the timer input used with channel 1 is selected by using TIS0 register).

```

;|||

```



```

;||||||||| [4]Timer interrupt is generated
;||||||||| when counting is started
;||||||||| (timer output also changes).
;|||||||||
;||||||||| : 0 1 1 0 : [1]Event counter mode
;||||||||| [2]Counting down
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is not generated
;||||||||| when counting is started
;||||||||| (timer output does not change, either).
;|||||||||
;||||||||| : 1 0 0 0 : [1]One-count mode
;||||||||| [2]Counting down
;||||||||| [3]Impossible
;||||||||| [4]Start trigger is invalid
;||||||||| during counting operation.
;||||||||| At that time, interrupt
;||||||||| is not generated, either.
;|||||||||
;||||||||| : 1 0 0 1 : [1]One-count mode
;||||||||| [2]Counting down
;||||||||| [3]Impossible
;||||||||| [4]Start trigger is valid
;||||||||| during counting operation.
;||||||||| At that time, interrupt
;||||||||| is also generated.
;|||||||||
;||||||||| : 1 1 0 0 : [1]Capture & one-count mode
;||||||||| [2]Counting up
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is not generated
;||||||||| when counting is started
;||||||||| (timer output does not change, either).
;||||||||| Start trigger is invalid
;||||||||| during counting operation.
;||||||||| At that time interrupt
;||||||||| is not generated, either.
;|||||||||
;||||||||| : Other than above : Setting prohibited
;|||||||||
;|||||||||++----- : Be sure to set 00
;|||||||||
;|||||||||++----- : Selection of TI06 pin input signal,

```

```

;|||||||          fSUB/2, fSUB/4, or INTRTC1 valid edge
;|||||||          (the timer input used with channel 6
;|||||||          is selected by using TIS0 register).
;||||||| : 0 0 : Falling edge
;||||||| : 0 1 : Rising edge
;||||||| : 1 0 : Both edges (when low-level width is measured)
;|||||||          Start trigger: Falling edge, Capture trigger: Rising edge
;||||||| : 1 1 : Both edges (when high-level width is measured)
;|||||||          Start trigger: Rising edge, Capture trigger: Falling edge
;|||||||
;|||||+++----- : Setting of start trigger or capture trigger of channel
6
;||||| : 0 0 0 : Only software trigger start is valid (other trigger sources
are unselected).
;||||| : 0 0 1 : Valid edge of TI06 pin input signal, fSUB/2, fSUB/4, or INTRTC1
is used as both the start trigger and capture trigger.
;||||| : 0 1 0 : Both the edges of TI06 pin input signal, fSUB/2, fSUB/4, or
INTRTC1 are used as a start trigger and a capture trigger.
;||||| : 1 0 0 : Interrupt signal of the master channel is used (when the channel
is used as a slave channel with the combination operation function).
;||||| : Other than above : Setting prohibited
;|||||
;|||||+----- : Selection of slave/master of channel 6
;||||| : 0 : Operates as slave channel with combination operation function.
;||||| : 1 : Operates as master channel with combination operation function.
;|||||
;|||+----- : Selection of count clock (TCLK) of channel 0
;||| : 0 : Operation clock MCK specified by CKS06 bit
;||| : 1 : Valid edge of input signal input from TI06 pin, fSUB/2, fSUB/4, or
INTRTC1
;|||          (the timer input used with channel 6 is selected by using TIS0
register).
;|||
;|++----- : Be sure to set 00
;|
;+----- : Selection of operation clock (MCK) of channel 6
; : 0 : Operation clock CK00 set by TPS0 register
; : 1 : Operation clock CK01 set by TPS0 register
MOVW    AX,      #1000010000001001B ; Timer Mode Register 07
MOVW    TMR07,AX;|||||||+++---- : [1]Operation mode of channel 7
;|||||||          [2]Count operation of TCR
;|||||||          [3]Independent operation
;|||||||          [4]Setting of starting counting and interrupt

```

```

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

```



```

;||||||| : 1 0 0 1 : [1]One-count mode
;|||||||           [2]Counting down
;|||||||           [3]Impossible
;|||||||           [4]Start trigger is valid
;|||||||           during counting operation.
;|||||||           At that time, interrupt
;|||||||           is also generated.
;|||||||
;||||||| : 1 1 0 0 : [1]Capture & one-count mode
;|||||||           [2]Counting up
;|||||||           [3]Possible
;|||||||           [4]Timer interrupt is not generated
;|||||||           when counting is started
;|||||||           (timer output does not change, either).
;|||||||           Start trigger is invalid
;|||||||           during counting operation.
;|||||||           At that time interrupt
;|||||||           is not generated, either.
;|||||||
;||||||| : Other than above : Setting prohibited
;|||||||
;|||||||++----- : Be sure to set 00
;|||||||
;|||||||++----- : Selection of TI07 pin input signal,
;|||||||           fSUB/2, fSUB/4, or INTRTC1 valid edge
;|||||||           (the timer input used with channel 7
;|||||||           is selected by using TIS0 register).
;||||||| : 0 0 : Falling edge
;||||||| : 0 1 : Rising edge
;||||||| : 1 0 : Both edges (when low-level width is measured)
;|||||||           Start trigger: Falling edge, Capture trigger: Rising edge
;||||||| : 1 1 : Both edges (when high-level width is measured)
;|||||||           Start trigger: Rising edge, Capture trigger: Falling edge
;|||||||
;|||||+++----- : Setting of start trigger or capture trigger of channel
7
;||||| : 0 0 0 : Only software trigger start is valid (other trigger sources
are unselected).
;||||| : 0 0 1 : Valid edge of TI07 pin input signal, fSUB/2, fSUB/4, or INTRTC1
is used as both the start trigger and capture trigger.
;||||| : 0 1 0 : Both the edges of TI07 pin input signal, fSUB/2, fSUB/4, or
INTRTC1 are used as a start trigger and a capture trigger.
;||||| : 1 0 0 : Interrupt signal of the master channel is used (when the channel

```

is used as a slave channel with the combination operation function).

```

;|||| : Other than above : Setting prohibited
;||||
;||||+----- : Selection of slave/master of channel 7
;|||| : 0 : Operates as slave channel with combination operation function.
;|||| : 1 : Operates as master channel with combination operation function.
;||||
;||||+----- : Selection of count clock (TCLK) of channel 0
;||| : 0 : Operation clock MCK specified by CKS07 bit
;||| : 1 : Valid edge of input signal input from TI07 pin, fSUB/2, fSUB/4, or

```

INTRTC1

```

;||| (the timer input used with channel 7 is selected by using TIS0

```

register).

```

;|||
;|+----- : Be sure to set 00
;|
;+----- : Selection of operation clock (MCK) of channel 7
; : 0 : Operation clock CK00 set by TPS0 register
; : 1 : Operation clock CK01 set by TPS0 register

```

```

; CK01 = fCLK = 20MHz -> 0.05[us/cIk] * 0xFFF[count] = 0.20475ms

```

```

MOVW   TDR06,  #(OFFFH - 1)      ; set interval time
MOVW   TDR07,  #OFFFH           ; set initial data(100%)

MOVW   HL,     #LOWW TOMOL
CLR1   [HL].6                ; TM06:toggle mode
SET1   [HL].7                ; TM07:combination operation mode
MOVW   HL,     #LOWW TOLOL
CLR1   [HL].6                ; T006:set output activ level High
CLR1   [HL].7                ; T007:set output activ level High
MOVW   HL,     #LOWW TOOL
CLR1   [HL].6                ; T006:Low
SET1   [HL].7                ; T007:High
MOVW   HL,     #LOWW TOEOL
CLR1   [HL].6                ; T006:output disable
SET1   [HL].7                ; T007:output enable
CLR1   P3.3                  ; T007 output low
SET1   TMMK06                 ; disable interrupt
SET1   TMMK07                 ; disable interrupt
CLR1   TMIF06                 ; clear interrupt request flag
MOV    TSOL,    #0C0H         ; TAU0 CH6 & CH7 start

```

RET

```

;-----;
; Module:      SINITVR                      ;
; Description:  Setting of Voltage reference ;
;   parameter: --                          ;
;   return    : --                          ;
;-----;

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

```

```

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

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

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

; wait for settling time to 17msec
SET1    TSOL.1 ; start TAU0 CH1
CLR1    TMIF01 ; clear interrupt request flag
JINIVR100: ; wait for 17ms
NOP
BF      TMIF01, $JINIVR100 ; an interrupt at 17ms intervals generated ? , No
CLR1    TMIF01 ; clear interrupt request flag
SET1    TTOL.1 ; stop TAU0 CH1

RET

;-----;
; Module:      SINITADC ;
; Description: Setting of A/D converter ;
; parameter:  -- ;
; return :  -- ;
;-----;
SINITADC:
MOV     ADM, #01000011B ; A/D Converter Mode Register
;| | | | | +--- : Comparator operation control
;| | | | | : 0 : Stops comparator operation

```

```

;||||| : 1 : Enables comparator operation
;|||||
;|+++++--- : A/D Converter Mode Register (ADM) Conversion Time Selection
;|      [1]Mode
;|      [2]Conversion Time Selection
;|      [3]Conversion Clock (fAD)
;| : 0 0 0 0 0 : [1]Conversion mode 1 [2]240/fCLK [3]fCLK/12
;| : 0 0 1 0 0 : [1]Conversion mode 1 [2]160/fCLK [3]fCLK/8
;| : 0 1 0 0 0 : [1]Conversion mode 1 [2]120/fCLK [3]fCLK/6
;| : 0 1 1 0 0 : [1]Conversion mode 1 [2]100/fCLK [3]fCLK/5
;| : 0 0 0 0 1 : [1]Conversion mode 2 [2]240/fCLK [3]fCLK/12
;| : 0 0 1 0 1 : [1]Conversion mode 2 [2]160/fCLK [3]fCLK/8
;| : 0 1 0 0 1 : [1]Conversion mode 2 [2]120/fCLK [3]fCLK/6
;| : 0 1 1 0 1 : [1]Conversion mode 2 [2]100/fCLK [3]fCLK/5
;| : 0 0 0 1 0 : [1]Conversion mode 3 [2]300/fCLK [3]fCLK/12
;| : 0 0 1 1 0 : [1]Conversion mode 3 [2]200/fCLK [3]fCLK/8
;| : 0 1 0 1 0 : [1]Conversion mode 3 [2]150/fCLK [3]fCLK/6
;| : 0 1 1 1 0 : [1]Conversion mode 3 [2]125/fCLK [3]fCLK/5
;| : when fCLK is 20MHz, Other than above : Setting prohibited
;|
;|+----- : A/D conversion operation mode selection
;| : 0 : continuous conversion mode
;| : 1 : single-conversion mode
;|
;+----- : A/D conversion operation control
; : 0 : Stops conversion operation
; : 1 : Enables conversion operation

```

```

MOV      ADPC, #00000001B      ; A/D Port Configuration Register
;|||+++++--- : Analog input (A)/digital I/O (D) switching
;||| :      +----- ANI15/AVREFM/P157
;||| :      | +--+----- ANI10/P152 - ANI8/AMP2+/P150
;||| :      | | | +--+----- ANI7/AMP20/P27 - ANI0/AMPO-/P20
;||| : 0 0 0 0 0 : A A A A A A A A A A
;||| : 0 0 0 0 1 : A A A A A A A A A D
;||| : 0 0 0 1 0 : A A A A A A A A D D
;||| : 0 0 0 1 1 : A A A A A A A A D D D
;||| : 0 0 1 0 0 : A A A A A A A D D D D
;||| : 0 0 1 0 1 : A A A A A A A D D D D D
;||| : 0 0 1 1 0 : A A A A A A D D D D D D
;||| : 0 0 1 1 1 : A A A A A D D D D D D D
;||| : 0 1 0 0 0 : A A A A D D D D D D D D
;||| : 0 1 0 0 1 : A A A D D D D D D D D D

```

```

;||| : 0 1 0 1 0 : A A D D D D D D D D D D
;||| : 0 1 1 1 1 : A D D D D D D D D D D D
;||| : 1 0 0 0 0 : D D D D D D D D D D D D
;|||
;++++----- : Be sure to set 000

MOV     ADS,     #00000001B           ; Analog Input Channel Specification Register
;|||++++--- : Analog input channel
;||| : 0 0 0 0 : ANI0
;||| : 0 0 0 1 : ANI1
;||| : 0 0 1 0 : ANI2
;||| : 0 0 1 1 : ANI3
;||| : 0 1 0 0 : ANI4
;||| : 0 1 0 1 : ANI5
;||| : 0 1 1 0 : ANI6
;||| : 0 1 1 1 : ANI7
;||| : 1 0 0 0 : ANI8
;||| : 1 0 0 1 : ANI9
;||| : 1 0 1 0 : ANI10
;||| : 1 1 1 1 : ANI15
;||| : Other than the above : Setting prohibited
;|||
;++++----- : Be sure to set 0000

MOV     ADM1,   #00000000B           ; A/D Converter Mode Register 1
;|||+--- : Selection of timer trigger signal
;||| : 0 : INTTM02
;||| : 1 : INTTM03
;|||
;|++++--- : Be sure to set 00000
;|
;+----- : Selection of A/D converter trigger mode
; : 0 : Software trigger mode
; : 1 : Timer trigger signal mode (hardware trigger mode)

SET1    ADMK           ; disable A/D interrupt

RET

end

```

```
main.c (C言語版)

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

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

/*-----*/
/* Include files
                                     */
/*-----*/
/*-----*/
/* Function prototyps
                                     */
/*-----*/
static void  fn_InitPort(void); /* Setting of I/O ports */
static void  fn_InitLvi(void); /* Low-voltage detection */
static void  fn_InitTau0(void); /* Setting of Timer array unit 0 */
static void  fn_InitVr(void); /* Setting of Voltage reference */
static void  fn_InitAdc(void); /* Setting of A/D converter */

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

```

                                                                    */
/*-----*/
#define          CGETAD          10          /* get A/D result number of times */

/*****/
/* data for calculation to convert          */
/* [12-bit A/D conversion result]          */
/* into [temperature(degrees celsius) * 100]. */
/*****/
/* Legend symbol
*   AD:   12-bit A/D conversion result
*   VAIN: Analog input of the A/D converter
*         = Output voltage of temperature sensor
*   C:    Temperature (degrees celsius)
*
* -----
* Features of temperature sensor (S-8120C)
*   * Linear output voltage for temperature changes.
*   * Temperature sensitivity : -8.2[mV/C] (Typ.)
*   * Output voltage : Ta = 30[C] -> 1.474[V] (Typ.)
* -----
*
* -> VAIN = -0.0082 * C + 1.72
*
*           (1.72 - VAIN) * 10000      8600 - (5000 * VAIN)
*   C = ----- = -----
*           82                        41
*
* -----
* Setting of A/D converter
*   * AVREF : 2.0[V]
*   * 12-bit resolution (result : 000H-FFFH)
* -----
*
* ->  AD = INT((VAIN/AVREF)*4096)
*     VAIN = (AD / 4096) * AVREF
*         = AD / 2048
*
*           8600 - 5000 * (AD / 2048)      2201600 - 625 * AD
*   C = ----- = -----
*           41                        10496
*
*           2201600 - 625 * AD          55040000 - 15625 * AD

```



```

    * C * 100 = ----- * 100 = -----
    *                10496                2624
    */
#define          CCALC1 55040000 /*          CCALC1 - CCALC2 * AD */
#define          CCALC2 15625          /* C * 100 = ----- */
#define          CCALC3 2624          /*          CCALC3          */

/*-----*/
/* Global variables
*/
/*-----*/
/*-----*/
/* Local variables
*/
/*-----*/
static unsigned short ushADBuffer[CGETAD]; /* Sum of A/D result */
static unsigned char ucSave; /* Save position of A/D result */
static unsigned short ushADAverage; /* Average of A/D result */
static unsigned char ucSign; /* sign of temperature(0:+,1:-) */
#define          CPLUS          0          /* + */
#define          CMINUS         1          /* - */
static unsigned short ushTemperature; /* Temperature * 100 @ HEX */
static boolean bFirstAd; /* first A/D conversion flag */

/*-----*/
/* Code
*/
/*-----*/
/*-----*/
/* Hardware initialization
*/
/*-----*/
void hdwinit(void)
{
    DI(); /* disable all interrupts */

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

/*-----*/
/*          Low-voltage detection          */

```

```

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

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

```

```

/*          [3]External clock input */

MSTOP = 0; /* X1 oscillator operating */

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

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

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

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

```

```

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

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

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

PER0 =    0b00100001;    /* Peripheral Enable Register 0 */
/*||||||+--- : Control of timer array unit 0 input clock */
/*|||||| : 0 : Stops input clock supply. */
/*||||||      * SFR used by timer array unit 0 cannot be written. */
/*||||||      * Timer array unit 0 is in the reset status. */
/*|||||| : 1 : Supplies input clock. */
/*||||||      * SFR used by timer array unit 0 can be read and written. */
/*|||||| */

```

```

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

```

```

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

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

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

    /*-----*/
    /*      Initialization of A/D converter      */
    /*-----*/
    fn_InitAdc();
}

/*-----*/
/* Module:      fn_InitPort
                */
/* Description:  Setting of I/O ports
                */
/*      parameter: --
                */
/*      return  : --
                */
/*-----*/
static void    fn_InitPort(void)
{
    /*-----*/
    /*      Setting of Port 0
                */
    /*-----*/
    P0 =      0b00000000;    /* Set P00-P02 Output latch to Low */
    PM0 =     0b11111000;    /* Set P00-P02 to output port */
                          /* P00-P02:Unused */

```

```

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

/*-----*/
/*      Setting of Port 2
          */
/*-----*/
P2 =      0b00000000;      /* Set P20-P27 Output latch to Low */
PM2 =     0b11111110;      /* Set P20 to output port, P21-P27 to input port */
                          /* P21:Analog input for sensor(ANI1) */
                          /* P20,P22-P27:Unused */

/*-----*/
/*      Setting of Port 3
          */
/*-----*/
P3 =      0b00001000;      /* Set P30-P32,P34 Output latch to Low */
                          /* Set P33 Output latch High */
PM3 =     0b11100000;      /* Set P30-P34 to output port */
                          /* P33:LED controlled by PWM output(TO07) */
                          /* P30-P32,P34:Unused */

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

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

```

```

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

/*-----*/
/*      Setting of Port 7
          */
/*-----*/
P7 =      0b00000000;    /* Set P70-P77 Output latch to Low */
PM7 =     0b00000000;    /* Set P70-P77 to output port */
                      /* P70-P77:Unused */

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

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

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

/*-----*/
/*      Setting of Port 11
          */

```



```

/*-----*/
    P11 =      0b00000000;      /* Set P110-P111 Output latch to Low */
    PM11 =     0b11111100;      /* Set P110-P111 to output port */
                                /* P110-P111:Unused */

/*-----*/
/*      Setting of Port 12
        */

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

/*-----*/
/*      Setting of Port 13
        */

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

/*-----*/
/*      Setting of Port 14
        */

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

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

/*-----*/
    P15 =      0b00000000;      /* Set P150-P152,P157 Output latch to Low */
    PM15 =     0b11111111;      /* Set P150-P152,P157 to input port */
                                /* P150-P152,P157:Unused */
}

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

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

```

```

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

    /* setting of Low-Voltage Detector */
    LVIMK      = 1;      /* disable LVI interrupt */
    LVISEL      = 0;      /* detects level of VDD */
    LVIS =      0b00001001;      /* Low-Voltage Detection Level Select Register */
    /*||||+++++--- : Detection level */
    /*|||| : 0 0 0 0 : VLVI0 (4.22V) */
    /*|||| : 0 0 0 1 : VLVI1 (4.07V) */
    /*|||| : 0 0 1 0 : VLVI2 (3.92V) */
    /*|||| : 0 0 1 1 : VLVI3 (3.76V) */
    /*|||| : 0 1 0 0 : VLVI4 (3.61V) */
    /*|||| : 0 1 0 1 : VLVI5 (3.45V) */
    /*|||| : 0 1 1 0 : VLVI6 (3.30V) */
    /*|||| : 0 1 1 1 : VLVI7 (3.15V) */
    /*|||| : 1 0 0 0 : VLVI8 (2.99V) */
    /*|||| : 1 0 0 1 : VLVI9 (2.84V) */
    /*|||| : 1 0 1 0 : VLVI10 (2.68V) */
    /*|||| : 1 0 1 1 : VLVI11 (2.53V) */
    /*|||| : 1 1 0 0 : VLVI12 (2.38V) */
    /*|||| : 1 1 0 1 : VLVI13 (2.22V) */
    /*|||| : 1 1 1 0 : VLVI14 (2.07V) */
    /*|||| : 1 1 1 1 : VLVI15 (1.91V) */
    /*|||| */
    /*+++++----- : Be sure to set 0000 */
    LVIMD      = 0;      /* generates an internal interrupt signal when detect the low-voltage */
    LVION      = 1;      /* enables low-voltage detection operation */

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

    /* wait for VDD to become VLVI or more */
    while(LVIF){
        NOP();
    }
}

```

```

    LVION      = 0;      /* disables low-voltage detection operation */
}

/*-----*/
/* Module:      fn_InitTau0
                */
/* Description:  Setting of Timer array unit 0          */
/* parameter:  --
                */
/* return   :  --
                */
/*-----*/
static void    fn_InitTau0(void)
{
    TPSOL =    0b00001000;      /* Timer Clock Select Register 0 */
    /*|||++++--- : Selection of operation clock (CK00) */
    /*++++----- : Selection of operation clock (CK01) */
    /* : 0 0 0 0 : CK0m = fCLK */
    /* : 0 0 0 1 : CK0m = fCLK/2 */
    /* : 0 0 1 0 : CK0m = fCLK/2^2 */
    /* : 0 0 1 1 : CK0m = fCLK/2^3 */
    /* : 0 1 0 0 : CK0m = fCLK/2^4 */
    /* : 0 1 0 1 : CK0m = fCLK/2^5 */
    /* : 0 1 1 0 : CK0m = fCLK/2^6 */
    /* : 0 1 1 1 : CK0m = fCLK/2^7 */
    /* : 1 0 0 0 : CK0m = fCLK/2^8 */
    /* : 1 0 0 1 : CK0m = fCLK/2^9 */
    /* : 1 0 1 0 : CK0m = fCLK/2^10 */
    /* : 1 0 1 1 : CK0m = fCLK/2^11 */
    /* : 1 1 0 0 : CK0m = fCLK/2^12 */
    /* : 1 1 0 1 : CK0m = fCLK/2^13 */
    /* : 1 1 1 0 : CK0m = fCLK/2^14 */
    /* : 1 1 1 1 : CK0m = fCLK/2^15 */
    /* m = 0, 1 */

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

```

```

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

```

```

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

```

```

/*||||| with the combination operation function). */
/*||||| : Other than above : Setting prohibited */
/*||||| */
/*|||||+----- : Selection of slave/master of channel 0 */
/*||||| : 0 : Operates as slave channel with combination operation function. */
/*||||| : 1 : Operates as master channel with combination operation function. */
/*||||| */
/*|||||+----- : Selection of count clock (TCLK) of channel 0 */
/*||||| : 0 : Operation clock MCK specified by CKS00 bit */
/*||||| : 1 : Valid edge of input signal input from TI00 pin, fSUB/2, fSUB/4, or INTRTC1 */
/*||||| (the timer input used with channel 0 is selected by using TIS0 register). */
/*||||| */
/*|||||+----- : Be sure to set 00 */
/*| */
/*+----- : Selection of operation clock (MCK) of channel 0 */
/* : 0 : Operation clock CK00 set by TPS0 register */
/* : 1 : Operation clock CK01 set by TPS0 register */

/* CK00 = fCLK/2^8 = 78.1kHz -> 12.804[us/clk] * 39050[count] = 499.996ms */
TDR00 = (39050 - 1); /* set interval time to about 500 ms */

TMMK00 = 0; /* enable interrupt */
TSOL.0 = 1; /* start TAU0 CH0 */
TMIF00 = 1; /* set interrupt request flag */

/*-----*/
/* CH1:for VR settling time */
/*-----*/
TMR01 = 0b0000000000000000; /* Timer Mode Register 01 */
/*|||||+----- : [1]Operation mode of channel 1 */
/*||||| [2]Count operation of TCR */
/*||||| [3]Independent operation */
/*||||| [4]Setting of starting counting and interrupt */
/*||||| : 0 0 0 0 : [1]Interval timer mode */
/*||||| [2]Counting down */
/*||||| [3]Possible */
/*||||| [4]Timer interrupt is not generated */
/*||||| when counting is started */
/*||||| (timer output does not change, either). */
/*||||| */
/*||||| : 0 0 0 1 : [1]Interval timer mode */
/*||||| [2]Counting down */
/*||||| [3]Possible */

```

```

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

```

```

/*|||||          [4]Timer interrupt is not generated */
/*|||||          when counting is started */
/*|||||          (timer output does not change, either). */
/*|||||          Start trigger is invalid */
/*|||||          during counting operation. */
/*|||||          At that time interrupt */
/*|||||          is not generated, either. */
/*||||| */
/*||||| : Other than above : Setting prohibited */
/*||||| */
/*|||||++----- : Be sure to set 00 */
/*||||| */
/*|||||++----- : Selection of TI01 pin input signal, */
/*|||||          fSUB/2, fSUB/4, or INTRTC1 valid edge */
/*|||||          (the timer input used with channel 1 */
/*|||||          is selected by using TIS0 register). */
/*||||| : 0 0 : Falling edge */
/*||||| : 0 1 : Rising edge */
/*||||| : 1 0 : Both edges (when low-level width is measured) */
/*|||||          Start trigger: Falling edge, Capture trigger: Rising edge */
/*||||| : 1 1 : Both edges (when high-level width is measured) */
/*|||||          Start trigger: Rising edge, Capture trigger: Falling edge */
/*||||| */
/*|||||+++----- : Setting of start trigger or capture trigger of channel 1 */
/*||||| : 0 0 0 : Only software trigger start is valid */
/*|||||          (other trigger sources are unselected). */
/*||||| : 0 0 1 : Valid edge of TI01 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          is used as both the start trigger and capture trigger. */
/*||||| : 0 1 0 : Both the edges of TI01 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          are used as a start trigger and a capture trigger. */
/*||||| : 1 0 0 : Interrupt signal of the master channel is used */
/*|||||          (when the channel is used as a slave channel */
/*|||||          with the combination operation function). */
/*||||| : Other than above : Setting prohibited */
/*||||| */
/*|||||+----- : Selection of slave/master of channel 1 */
/*||||| : 0 : Operates as slave channel with combination operation function. */
/*||||| : 1 : Operates as master channel with combination operation function. */
/*||||| */
/*|||||+----- : Selection of count clock (TCLK) of channel 0 */
/*||||| : 0 : Operation clock MCK specified by CKS01 bit */
/*||||| : 1 : Valid edge of input signal input from TI01 pin, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          (the timer input used with channel 1 is selected by using TIS0 register). */

```



```

/*||| */
/*|++----- : Be sure to set 00 */
/*| */
/*+----- : Selection of operation clock (MCK) of channel 1 */
/* : 0 : Operation clock CK00 set by TPS0 register */
/* : 1 : Operation clock CK01 set by TPS0 register */

/* CK00 = fCLK/2^8 = 78.1kHz -> 12.804[us/clock] * 1328[count] = 17.003ms */
TDR01 = (1328 - 1); /* set interval time to about 17 ms */

TMMK01 = 1; /* disable interrupt */

/*-----*/
/* CH6 & CH7:for PWM output */
/*-----*/
TMR06 = 0b1000100000000001; /* Timer Mode Register 06 */
/*|||++++ : [1]Operation mode of channel 6 */
/*||| [2]Count operation of TCR */
/*||| [3]Independent operation */
/*||| [4]Setting of starting counting and interrupt */
/*||| : 0 0 0 0 : [1]Interval timer mode */
/*||| [2]Counting down */
/*||| [3]Possible */
/*||| [4]Timer interrupt is not generated */
/*||| when counting is started */
/*||| (timer output does not change, either). */
/*||| */
/*||| : 0 0 0 1 : [1]Interval timer mode */
/*||| [2]Counting down */
/*||| [3]Possible */
/*||| [4]Timer interrupt is generated */
/*||| when counting is started */
/*||| (timer output also changes). */
/*||| */
/*||| : 0 1 0 0 : [1]Capture mode */
/*||| [2]Counting up */
/*||| [3]Possible */
/*||| [4]Timer interrupt is not generated */
/*||| when counting is started */
/*||| (timer output does not change, either). */
/*||| */
/*||| : 0 1 0 1 : [1]Capture mode */
/*||| [2]Counting up */

```

```

/*|||||||          [3]Possible */
/*|||||||          [4]Timer interrupt is generated */
/*|||||||          when counting is started */
/*|||||||          (timer output also changes). */
/*||||||| */
/*||||||| : 0 1 1 0 : [1]Event counter mode */
/*|||||||          [2]Counting down */
/*|||||||          [3]Possible */
/*|||||||          [4]Timer interrupt is not generated */
/*|||||||          when counting is started */
/*|||||||          (timer output does not change, either). */
/*||||||| */
/*||||||| : 1 0 0 0 : [1]One-count mode */
/*|||||||          [2]Counting down */
/*|||||||          [3]Impossible */
/*|||||||          [4]Start trigger is invalid */
/*|||||||          during counting operation. */
/*|||||||          At that time, interrupt */
/*|||||||          is not generated, either. */
/*||||||| */
/*||||||| : 1 0 0 1 : [1]One-count mode */
/*|||||||          [2]Counting down */
/*|||||||          [3]Impossible */
/*|||||||          [4]Start trigger is valid */
/*|||||||          during counting operation. */
/*|||||||          At that time, interrupt */
/*|||||||          is also generated. */
/*||||||| */
/*||||||| : 1 1 0 0 : [1]Capture & one-count mode */
/*|||||||          [2]Counting up */
/*|||||||          [3]Possible */
/*|||||||          [4]Timer interrupt is not generated */
/*|||||||          when counting is started */
/*|||||||          (timer output does not change, either). */
/*|||||||          Start trigger is invalid */
/*|||||||          during counting operation. */
/*|||||||          At that time interrupt */
/*|||||||          is not generated, either. */
/*||||||| */
/*||||||| : Other than above : Setting prohibited */
/*||||||| */
/*|||||||++----- : Be sure to set 00 */
/*||||||| */

```

```

/*|||||++----- : Selection of TI06 pin input signal, */
/*|||||          fSUB/2, fSUB/4, or INTRTC1 valid edge */
/*|||||          (the timer input used with channel 6 */
/*|||||          is selected by using TIS0 register). */
/*||||| : 0 0 : Falling edge */
/*||||| : 0 1 : Rising edge */
/*||||| : 1 0 : Both edges (when low-level width is measured) */
/*|||||          Start trigger: Falling edge, Capture trigger: Rising edge */
/*||||| : 1 1 : Both edges (when high-level width is measured) */
/*|||||          Start trigger: Rising edge, Capture trigger: Falling edge */
/*||||| */
/*||||++----- : Setting of start trigger or capture trigger of channel 6 */
/*|||| : 0 0 0 : Only software trigger start is valid */
/*||||          (other trigger sources are unselected). */
/*|||| : 0 0 1 : Valid edge of TI06 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*||||          is used as both the start trigger and capture trigger. */
/*|||| : 0 1 0 : Both the edges of TI06 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*||||          are used as a start trigger and a capture trigger. */
/*|||| : 1 0 0 : Interrupt signal of the master channel is used */
/*||||          (when the channel is used as a slave channel */
/*||||          with the combination operation function). */
/*|||| : Other than above : Setting prohibited */
/*|||| */
/*|||+----- : Selection of slave/master of channel 6 */
/*||| : 0 : Operates as slave channel with combination operation function. */
/*||| : 1 : Operates as master channel with combination operation function. */
/*||| */
/*||+----- : Selection of count clock (TCLK) of channel 0 */
/*|| : 0 : Operation clock MCK specified by CKS06 bit */
/*|| : 1 : Valid edge of input signal input from TI06 pin, fSUB/2, fSUB/4, or INTRTC1 */
/*||          (the timer input used with channel 6 is selected by using TIS0 register). */
/*|| */
/*|++----- : Be sure to set 00 */
/*| */
/*+----- : Selection of operation clock (MCK) of channel 6 */
/* : 0 : Operation clock CK00 set by TPS0 register */
/* : 1 : Operation clock CK01 set by TPS0 register */
TMR07 = 0b1000010000001001; /* Timer Mode Register 07 */
/*|||||||||++----- : [1]Operation mode of channel 7 */
/*|||||||||          [2]Count operation of TCR */
/*|||||||||          [3]Independent operation */
/*|||||||||          [4]Setting of starting counting and interrupt */
/*||||||||| : 0 0 0 0 : [1]Interval timer mode */

```

```

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

```

```

/*|||||||          [2]Counting down */
/*|||||||          [3]Impossible */
/*|||||||          [4]Start trigger is valid */
/*|||||||          during counting operation. */
/*|||||||          At that time, interrupt is also generated. */
/*||||||| */
/*||||||| : 1 1 0 0 : [1]Capture & one-count mode */
/*|||||||          [2]Counting up */
/*|||||||          [3]Possible */
/*|||||||          [4]Timer interrupt is not generated */
/*|||||||          when counting is started */
/*|||||||          (timer output does not change, either). */
/*|||||||          Start trigger is invalid */
/*|||||||          during counting operation. */
/*|||||||          At that time interrupt */
/*|||||||          is not generated, either. */
/*||||||| */
/*||||||| : Other than above : Setting prohibited */
/*||||||| */
/*|||||||++----- : Be sure to set 00 */
/*||||||| */
/*|||||||++----- : Selection of TI07 pin input signal, */
/*|||||||          fSUB/2, fSUB/4, or INTRTC1 valid edge */
/*|||||||          (the timer input used with channel 7 */
/*|||||||          is selected by using TIS0 register). */
/*||||||| : 0 0 : Falling edge */
/*||||||| : 0 1 : Rising edge */
/*||||||| : 1 0 : Both edges (when low-level width is measured) */
/*|||||||          Start trigger: Falling edge, Capture trigger: Rising edge */
/*||||||| : 1 1 : Both edges (when high-level width is measured) */
/*|||||||          Start trigger: Rising edge, Capture trigger: Falling edge */
/*||||||| */
/*|||||+++----- : Setting of start trigger or capture trigger of channel 7 */
/*||||| : 0 0 0 : Only software trigger start is valid */
/*|||||          (other trigger sources are unselected). */
/*||||| : 0 0 1 : Valid edge of TI07 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          is used as both the start trigger and capture trigger. */
/*||||| : 0 1 0 : Both the edges of TI07 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          are used as a start trigger and a capture trigger. */
/*||||| : 1 0 0 : Interrupt signal of the master channel is used */
/*|||||          (when the channel is used as a slave channel */
/*|||||          with the combination operation function). */
/*||||| : Other than above : Setting prohibited */

```

```

/*|||| */
/*||||+----- : Selection of slave/master of channel 7 */
/*|||| : 0 : Operates as slave channel with combination operation function. */
/*|||| : 1 : Operates as master channel with combination operation function. */
/*|||| */
/*|||+----- : Selection of count clock (TCLK) of channel 0 */
/*||| : 0 : Operation clock MCK specified by CKS07 bit */
/*||| : 1 : Valid edge of input signal input from TI07 pin, fSUB/2, fSUB/4, or INTRTC1 */
/*||| (the timer input used with channel 7 is selected by using TIS0 register). */
/*||| */
/*|++----- : Be sure to set 00 */
/*| */
/*+----- : Selection of operation clock (MCK) of channel 7 */
/* : 0 : Operation clock CK00 set by TPS0 register */
/* : 1 : Operation clock CK01 set by TPS0 register */

/* CK01 = fCLK = 20MHz -> 0.05[us/clock] * 0xFFFF[count] = 0.20475ms */
TDR06 = (0x0FFF - 1); /* set interval time */
TDR07 = 0x0FFF; /* set initial data (100%) */

TOM0L &= 0xbf; /* TM06:toggle mode */
TOM0L |= 0x80; /* TM07:combination operation mode */
TOL0L &= 0x3f; /* set output activ level High */
TO0L &= 0x0bf; /* TO06:Low */
TO0L |= 0x080; /* TO07:High */
TOE0L &= 0xbf; /* TO06:out disable */
TOE0L |= 0x80; /* TO07:out enable */
P3.3 = 0; /* TO07 output low */
TMMK06 = 1; /* disable interrupt */
TMMK07 = 1; /* disable interrupt */
TMIF06 = 0; /* clear interrupt request flag */
TSOL = 0x0c0; /* TAU0 CH6 & CH7 start */
}

/*-----*/
/* Module: fn_InitVr
*/
/* Description: Setting of Voltage reference */
/* parameter: --
*/
/* return : --
*/
/*-----*/

```

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

```

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

```

```

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

```

```

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

```

```

/* wait for settling time to 17msec */

```

```

TSOL.1 = 1; /* start TAU0 CH1 */

```

```

TMIF01 = 0;

```

```

while(!TMIF01){

```

```

    NOP();

```

```

}

```

```

TMIF01 = 0;

```

```

TTOL.1 = 1; /* stop TAU0 CH1 */

```

```

}

```

```

/*-----*/

```

```

/* Module:      fn_InitAdc
                */

```

```

/* Description:  Setting of A/D converter */

```

```

/* parameter: --
                */

```

```

/* return : --
                */

```

```

/*-----*/

```

```

static void fn_InitAdc(void)

```

```

{

```

```

    ADM = 0b01000011; /* A/D Converter Mode Register */

```

```

    /*||| |||+--- : Comparator operation control */

```

```

    /*||| ||| : 0 : Stops comparator operation */

```

```

    /*||| ||| : 1 : Enables comparator operation */

```

```

    /*||| ||| */

```

```

    /*|+++++----- : A/D Converter Mode Register (ADM) Conversion Time Selection */

```

```

    /*| [1]Mode */

```

```

    /*| [2]Conversion Time Selection */

```



```

/*||          [3]Conversion Clock (fAD) */
/*|| : 0 0 0 0 0 : [1]Conversion mode 1 [2]240/fCLK [3]fCLK/12 */
/*|| : 0 0 1 0 0 : [1]Conversion mode 1 [2]160/fCLK [3]fCLK/8 */
/*|| : 0 1 0 0 0 : [1]Conversion mode 1 [2]120/fCLK [3]fCLK/6 */
/*|| : 0 1 1 0 0 : [1]Conversion mode 1 [2]100/fCLK [3]fCLK/5 */
/*|| : 0 0 0 0 1 : [1]Conversion mode 2 [2]240/fCLK [3]fCLK/12 */
/*|| : 0 0 1 0 1 : [1]Conversion mode 2 [2]160/fCLK [3]fCLK/8 */
/*|| : 0 1 0 0 1 : [1]Conversion mode 2 [2]120/fCLK [3]fCLK/6 */
/*|| : 0 1 1 0 1 : [1]Conversion mode 2 [2]100/fCLK [3]fCLK/5 */
/*|| : 0 0 0 1 0 : [1]Conversion mode 3 [2]300/fCLK [3]fCLK/12 */
/*|| : 0 0 1 1 0 : [1]Conversion mode 3 [2]200/fCLK [3]fCLK/8 */
/*|| : 0 1 0 1 0 : [1]Conversion mode 3 [2]150/fCLK [3]fCLK/6 */
/*|| : 0 1 1 1 0 : [1]Conversion mode 3 [2]125/fCLK [3]fCLK/5 */
/*|| : when fCLK is 20MHz, Other than above : Setting prohibited */
/*|| */
/*|+----- : A/D conversion operation mode selection */
/*| : 0 : continuous conversion mode */
/*| : 1 : single-conversion mode */
/*| */
/*+----- : A/D conversion operation control */
/* : 0 : Stops conversion operation */
/* : 1 : Enables conversion operation */

```

```

ADPC = 0b00000001; /* A/D Port Configuration Register */
/*|||+++++--- : Analog input (A)/digital I/O (D) switching */
/*||| : +----- ANI15/AVREFM/P157 */
/*||| : | +-+----- ANI10/P152 - ANI8/AMP2+/P150 */
/*||| : ||| +-+----- ANI7/AMP2O/P27 - ANI0/AMP0-/P20 */
/*||| : 0 0 0 0 0 : A A A A A A A A A A A A */
/*||| : 0 0 0 0 1 : A A A A A A A A A A A D */
/*||| : 0 0 0 1 0 : A A A A A A A A A A D D */
/*||| : 0 0 0 1 1 : A A A A A A A A A D D D */
/*||| : 0 0 1 0 0 : A A A A A A A A D D D D */
/*||| : 0 0 1 0 1 : A A A A A A A D D D D D */
/*||| : 0 0 1 1 0 : A A A A A A D D D D D D */
/*||| : 0 0 1 1 1 : A A A A A D D D D D D D */
/*||| : 0 1 0 0 0 : A A A A D D D D D D D D */
/*||| : 0 1 0 0 1 : A A A D D D D D D D D D */
/*||| : 0 1 0 1 0 : A A D D D D D D D D D D */
/*||| : 0 1 1 1 1 : A D D D D D D D D D D D */
/*||| : 1 0 0 0 0 : D D D D D D D D D D D D */
/*||| */
/*+---+----- : Be sure to set 000 */

```

```

ADS =      0b00000001;      /* Analog Input Channel Specification Register */
/*||||+---- : Analog input channel */
/*|||| : 0 0 0 0 : ANI0 */
/*|||| : 0 0 0 1 : ANI1 */
/*|||| : 0 0 1 0 : ANI2 */
/*|||| : 0 0 1 1 : ANI3 */
/*|||| : 0 1 0 0 : ANI4 */
/*|||| : 0 1 0 1 : ANI5 */
/*|||| : 0 1 1 0 : ANI6 */
/*|||| : 0 1 1 1 : ANI7 */
/*|||| : 1 0 0 0 : ANI8 */
/*|||| : 1 0 0 1 : ANI9 */
/*|||| : 1 0 1 0 : ANI10 */
/*|||| : 1 1 1 1 : ANI15 */
/*|||| : Other than the above : Setting prohibited */
/*|||| */
/*++++----- : Be sure to set 0000 */

ADM1 =     0b00000000;     /* A/D Converter Mode Register 1 */
/*|||||+--- : Selection of timer trigger signal */
/*||||| : 0 : INTTM02 */
/*||||| : 1 : INTTM03 */
/*||||| */
/*|+++++---- : Be sure to set 000000 */
/*| */
/*+----- : Selection of A/D converter trigger mode */
/* : 0 : Software trigger mode */
/* : 1 : Timer trigger signal mode (hardware trigger mode) */

ADMK = 1; /* disable A/D interrupt */
}

/*-----*/
/* Module:      main
*/
/* Description:  Main process
*/
/* parameter: --
*/
/* return  : --
*/
/*-----*/

```

```
void main(void)
{
    unsigned char loop;    /* waiting counter */
    unsigned long int work; /* work area */

    /*-----*/
    /* Initialization of variable */
    /*-----*/
    ushTemperature = 0;    /* initialize temperature value */
    ucSign = CPLUS;       /* initialize sign of temperature */
    ucSave = 0;           /* initialize save position of A/D result */
    ushADAverage = 0;     /* initialize average */
    bFirstAd = 1;         /* set A/D conversion flag */

    /*=====*/
    /*-----*/
    /* Main Loop */
    /*-----*/
    /*=====*/
    while(1){

        HALT(); /* wait 500ms & cut electric power consumption */

        if(TMIF00){
            TMIF00 = 0;

            /*-----*/
            /* get A/D conversion result */
            /*-----*/
            ADCS = 1; /* A/D converter enable */
            ADIF = 0; /* clear A/D interrupt request flag */
            ADMK = 0; /* enable A/D interrupt */
            HALT(); /* set HALT mode and wait conversion finished */
            ADMK = 1; /* disable A/D interrupt */
            ADIF = 0; /* clear A/D interrupt request flag */

            ushADBuffer[ucSave] = ADCR; /* get A/D result */

            if(bFirstAd){ /* if first A/D conversion */
                bFirstAd = 0; /* clear A/D conversion flag */
                for(loop = 1; loop < CGETAD; loop++){
                    ushADBuffer[loop] = ushADBuffer[0]; /* initialize buffer */
                }
            }
        }
    }
}
```

```

}

ucSave = ((ucSave + 1) % CGETAD);          /* A/D result save position update */

/*-----*/
/* calculate A/D result average          */
/*-----*/
for(loop = 0, work = 0; loop < CGETAD; loop++){
    work += ushADBuffer[loop];
}
ushADAverage = (unsigned short)(work / CGETAD); /* average of A/D result */

/*-----*/
/* calculate temperature(degrees celsius) * 100 */
/*
/*          CCALC1 - CCALC2 * ushADAverage */
/* ushTemperature = ----- */
/*          CCALC3          */
/*-----*/
work = CCALC2 * (unsigned long int)ushADAverage;
if(work > CCALC1){
    ucSign = CMINUS; /* -nn.nn degrees celsius */
    work = work - CCALC1;
}
else{
    ucSign = CPLUS; /* nnn.nn degrees celsius */
    work = CCALC1 - work;
}
ushTemperature = (unsigned short)(work/CCALC3); /* save temperature * 100 */

/* adjust [-0.00 degrees] to [0.00 degrees] */
if(ushTemperature == 0){
    ucSign = CPLUS; /* set sign plus */
}

/*****/
/* The calculation of the temperature was completed. */
/* Please refer to ucSign(+/-) & ushTemperature(temperature * 100). */
/*****/

/*-----*/
/* PWM output update */
/*-----*/

```

```

        TDR07 = ushADAverage;
    }

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

}
}

```

付録B 改版履歴

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

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

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