

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

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

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

2010年4月1日  
ルネサスエレクトロニクス株式会社

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

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


## ご注意書き

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

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

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

# アプリケーション・ノート

NECエレクトロニクス株式会社  
マイクロコンピュータ事業部  
製品ソリューション グループ  
グループ マネージャー  
大場 浩司   
(担当: 西浦 真平)

## 78K0R/Kx3-L

### サンプル・プログラム リアルタイム・カウンタ編

この資料は、サンプル・プログラムの動作概要や使用方法、およびRTC機能を使用する際の設定方法や活用方法を説明したものです。サンプル・プログラムでは、RTCを用いた定周期割り込み機能、インターバル割り込み機能、およびアラーム割り込み機能の実装例を示しています。

#### 対象デバイス

78K0R/KC3-L(44pin)  
マイクロコントローラ  
78K0R/KC3-L(48pin)  
マイクロコントローラ  
78K0R/KD3-Lマイクロコントローラ  
78K0R/KE3-Lマイクロコントローラ  
78K0R/KF3-Lマイクロコントローラ  
78K0R/KG3-Lマイクロコントローラ

#### 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	周辺ハードウェア	・・・5
第3章	ソフトウェアについて	・・・6
3.1	ファイル構成	・・・6
3.2	使用する内蔵周辺機能	・・・7
3.3	リアルタイム・カウンタ機能の設定と動作概要	・・・7
3.4	フロー・チャート	・・・8
第4章	設定方法について	・・・11
4.1	リアルタイム・カウンタの設定	・・・11
4.2	割り込みの設定	・・・11
4.3	リアルタイム・カウンタの設定レジスタ	・・・12
4.4	割り込みの設定レジスタ	・・・18
4.5	時計誤差補正について	・・・20
4.6	リアルタイム・カウンタ設定の初期設定概要	・・・22
4.7	RTC読み出し/書き込みを行う際の設定概要	・・・28
4.8	RTCアラームを使用する際の設定概要	・・・30
第5章	PM+を用いたHEXファイルの生成	・・・32
5.1	ダウンロードファイルの解説	・・・32
5.2	サンプル・プログラムのHEXファイル生成	・・・33
5.3	開発環境のダウンロード、インストール	・・・35
第6章	関連資料	・・・36
付録A	プログラム・リスト	・・・37
	・アセンブリ言語版	・・・37
	・C言語版	・・・55
付録B	改版履歴	・・・76

・本資料に記載されている内容は2009年1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、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。

(2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

# 第1章 概 要

このサンプル・プログラムではリアルタイム・カウンタ（RTC）の定周期割り込み機能、インターバル割り込み機能、およびアラーム割り込み機能の使用例を示しています。現在設定日時から30秒経過後にアラーム割り込みを発生させます。この期間中は定周期割り込み、インターバル割り込みそれぞれの周期により、外部出力端子への反転出力を行います。

## （1）初期設定の主な内容

- ウォッチドッグ・タイマの動作停止
- システム・クロック・ソースとして、高速内蔵発振器を選択
- サブシステム・クロック・ソースとして、XT1発振回路を選択
- CPU / 周辺ハードウェア・クロック周波数を8MHzに設定
- LVI動作設定
- 入出力ポートの設定<sup>注</sup>
  - ・ P11を出力ポートに設定（初期出力はロウ・レベル）
  - ・ P12を出力ポートに設定（初期出力はロウ・レベル）
  - ・ P13を出力ポートに設定（初期出力はロウ・レベル）
- RTCへの初期設定
  - ・ RTCへの入力クロックを供給
  - ・ RTC1Hz端子出力禁止
  - ・ RTCCL端子出禁止
  - ・ 時刻表現を24時間制
  - ・ RTC定周期割り込み許可設定
    - 周期1sでの定周期割り込みを使用
  - ・ RTCインターバル割り込み許可設定
    - 周期125msでのインターバル割り込みを使用
  - ・ 現在日時を初期化
    - 1/1/1（日） 00:00:00
  - ・ RTCの動作開始
- 割り込みの許可
  - ・ INTRTC割り込みの許可
  - ・ INTRTCI割り込みの許可

**注：未使用端子のポート設定は、各デバイスによって設定が異なります。以下の各デバイスのポートをLow出力オープンに設定してください。ポートの設定については、サンプル・プログラム初期設定編アプリケーション・ノートの「4.4 ポートの設定レジスタ」を参照して下さい。**

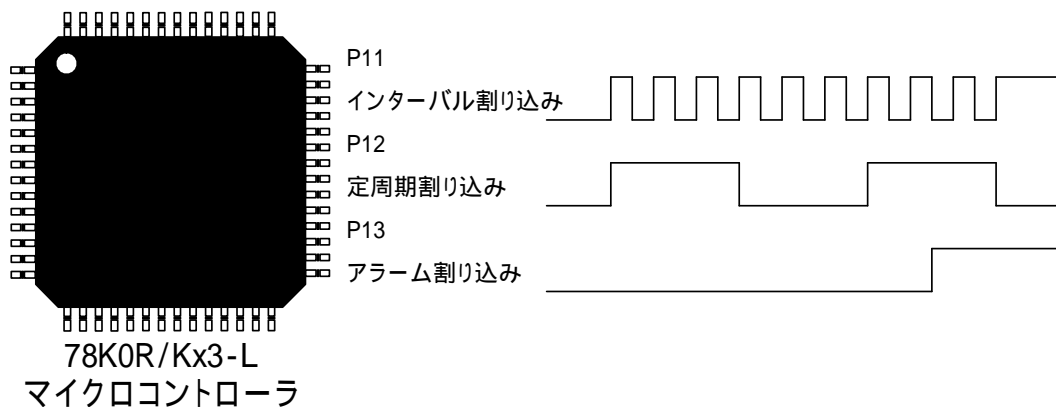
- ・ KC3-L(44pin)  
未使用端子：P10, P20-27, P30-32, P40-41, P50-52, P70-75, P80-83, P120, P150-151

- KC3-L(48pin)  
未使用端子：P10, P20-27, P30-32, P40-41, P50-51, P60-61, P70-75, P80-83, P120  
P140, P150-152
- KD3-L  
未使用端子：P00-01, P10, P20-27, P30-32, P40-41, P50-52, P60-61, P70-77, P80-83  
P120, P140, P150-152
- KE3-L  
未使用端子：P00-01, P10, P14-17, P20-27, P30-33, P40-43, P50-53, P60-61, P70-77  
P80-83, P120, P140-141, P150-153
- KF3-L  
未使用端子：P02-06, P10, P14-17, P20-27, P30-31, P40-47, P50-55, P60-67, P70-77  
P90-91, P110-111, P120, P130, P140, P142-144, P150-153
- KG3-L  
未使用端子：P00-06, P10, P14-17, P20-27, P30-31, P40-47, P50-57, P60-67, P70-77  
P80-87, P91, P110-111, P120, P130-131, P140-145, P150-157

(2) メイン・ループ以降の内容

初期設定完了後は、現在日時を 2008/4/10(SUN) 15:59:30 として設定します。続いてアラーム日時を 毎日 16:00:00 として設定し、次の割り込みによる各外部出力を行います。

- インターバル割り込み発生の場合は外部出力端子 (P11) への反転出力
- 定周期割り込み発生の場合は外部出力端子 (P12) への反転出力
- アラーム割り込み発生場合は外部出力端子 (P13) へハイ・レベル出力。



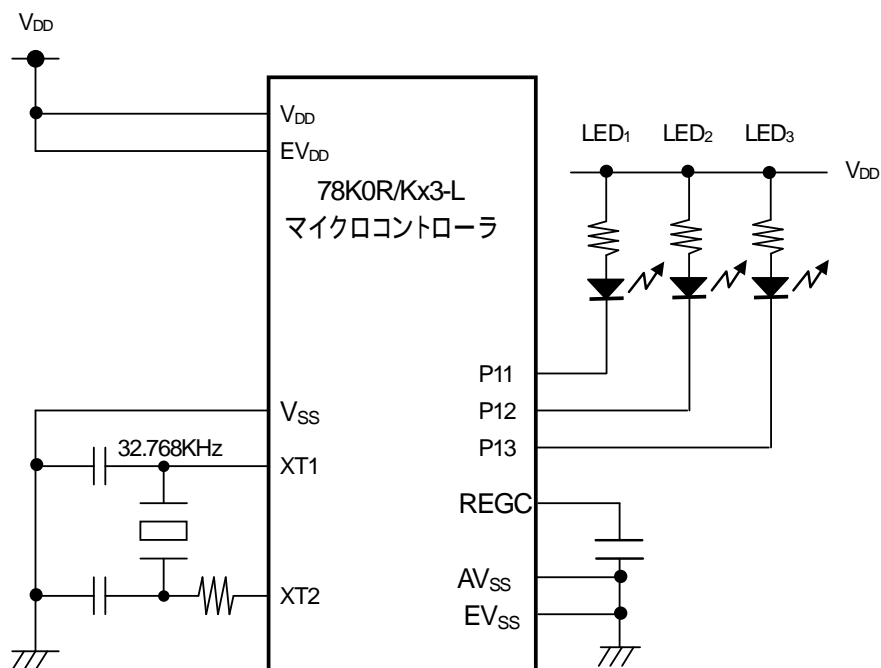
尚、デバイス使用上の注意事項については、[78K0R/Kx3-Lユーザーズ・マニュアル](#)を参照してください。

## 第2章 回路イメージ

この章では、このサンプル・プログラムで使用する回路イメージおよび周辺ハードウェアを説明します。

### 2.1 回路イメージ

サンプル・プログラムで使用する周辺ハードウェア回路イメージを次に示します。



注意：この回路イメージは接続の概要を示す為に簡略化しています。実際に回路を作成される場合は、端子処理などを適切に行い、電流など電気的特性を満たすように設計してください(P121 ~ P124は入力専用ポートになりますから個別に抵抗を介して $V_{DD}$ 又は $V_{SS}$ に接続して下さい)。

### 2.2 周辺ハードウェア

- ・ P11：インターバル割り込み発生（反転出力）
- ・ P12：定周期割り込み発生（反転出力）
- ・ P13：アラーム割り込み発生（ハイ・レベル出力）



## 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。



#### 【C言語版】


ファイル名	説明	同封圧縮 (*.zip) ファイル	
			
Kx3-L_RTC.c	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		
OP.asm	オプション・バイトの指定ファイル		
78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル	-	
78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ	-	


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

#### 【アセンブリ言語版】

ファイル名	説明	同封圧縮 (*.zip) ファイル	
			
Kx3-L_RTC.asm	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		
OP.asm	オプション・バイトの指定ファイル		
78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル	-	
78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ	-	

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



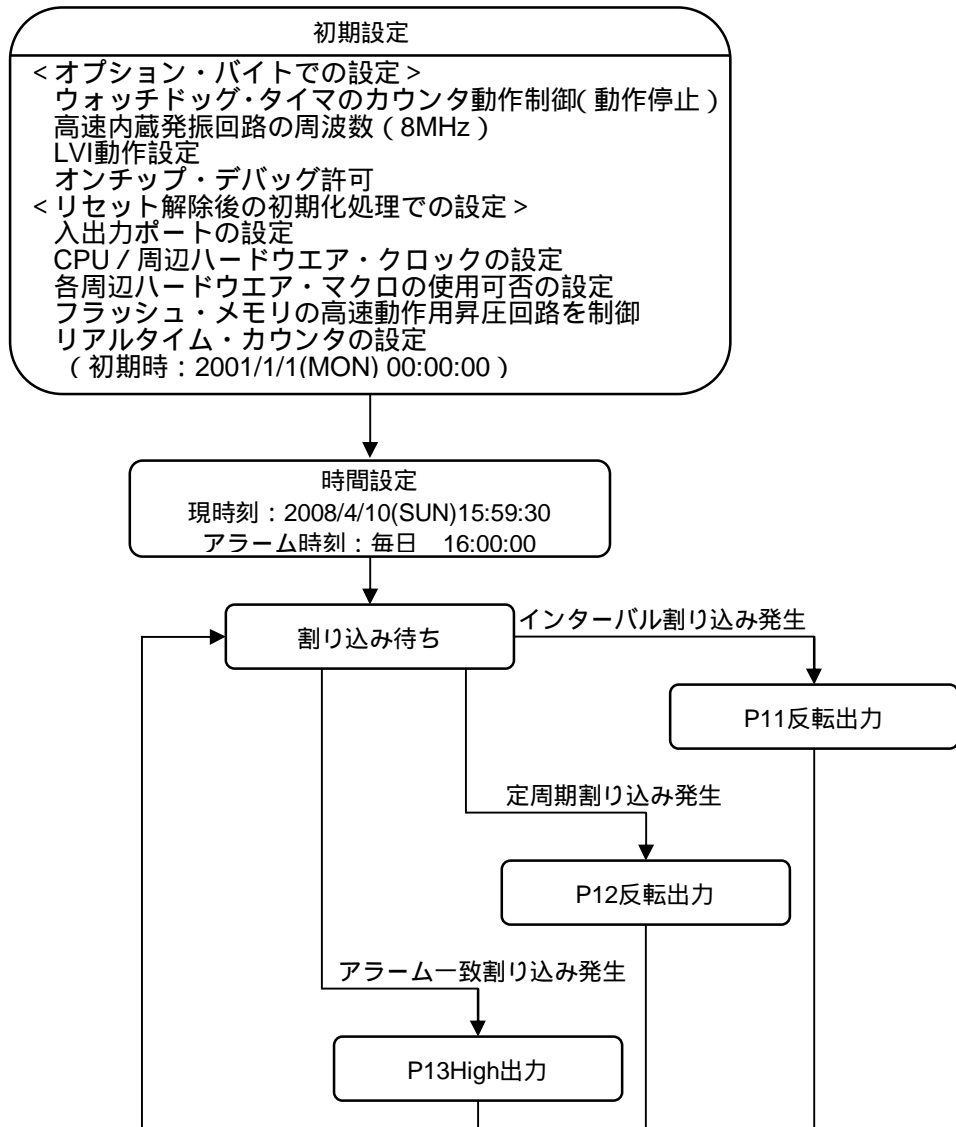
### 3.2 使用する内蔵周辺機能

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。

- ・リアルタイム・カウンタ : INTRTCI, INTRTC割り込みのハンドル
- ・LED1出力(インターバル割り込み) : P11
- ・LED2出力(定周期割り込み) : P12
- ・LED3出力(アラーム割り込み) : P13

### 3.3 リアルタイム・カウンタ機能の設定と動作概要

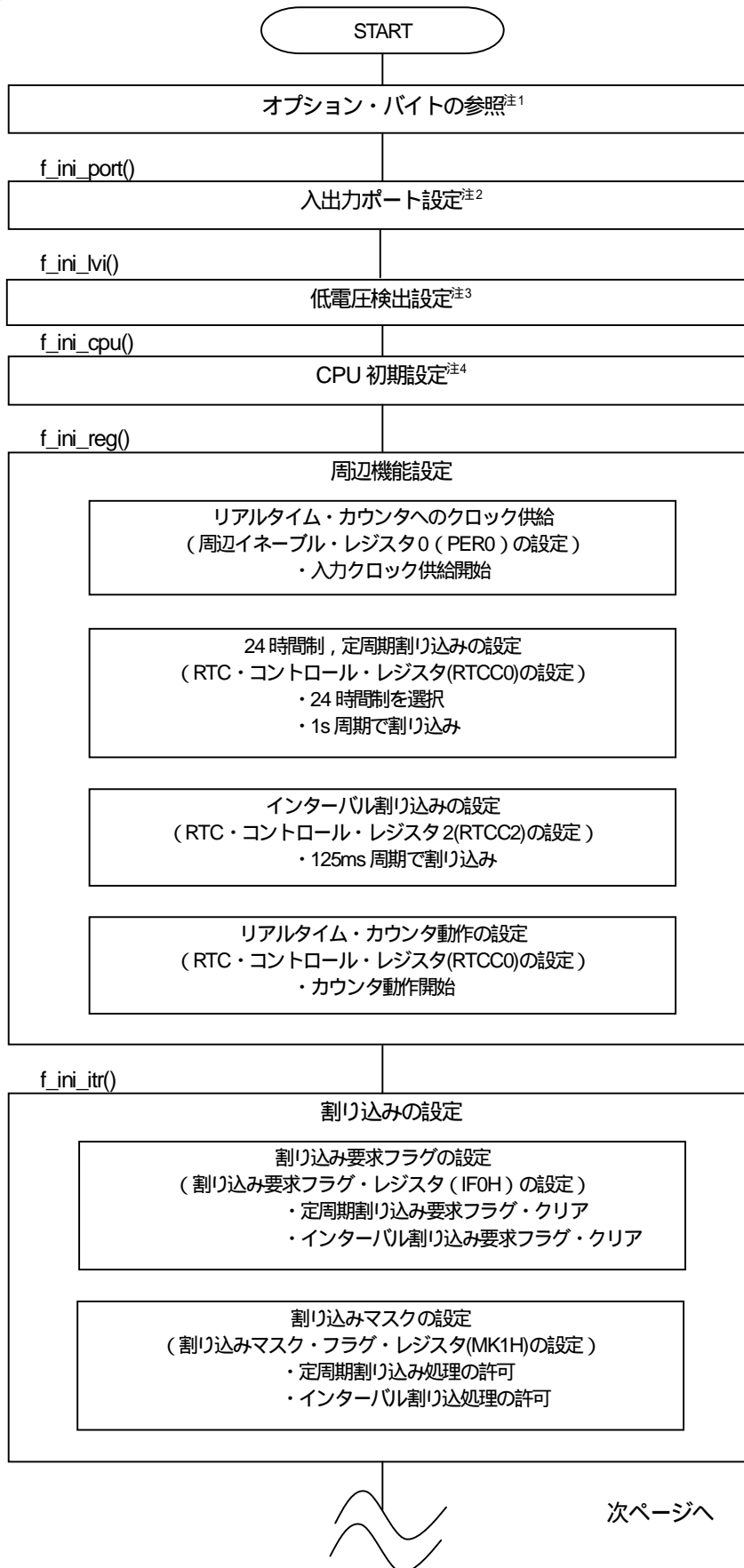
このサンプル・プログラムでは、初期設定にてクロック周波数の選択, 入出力ポートの設定, RTCの設定(周期0.5sでの定周期割り込み, 周期125msでのインターバル割り込みを使用, 現在日時情報の初期化)の設定などを行います。処理詳細については、以下の状態遷移図に示します。

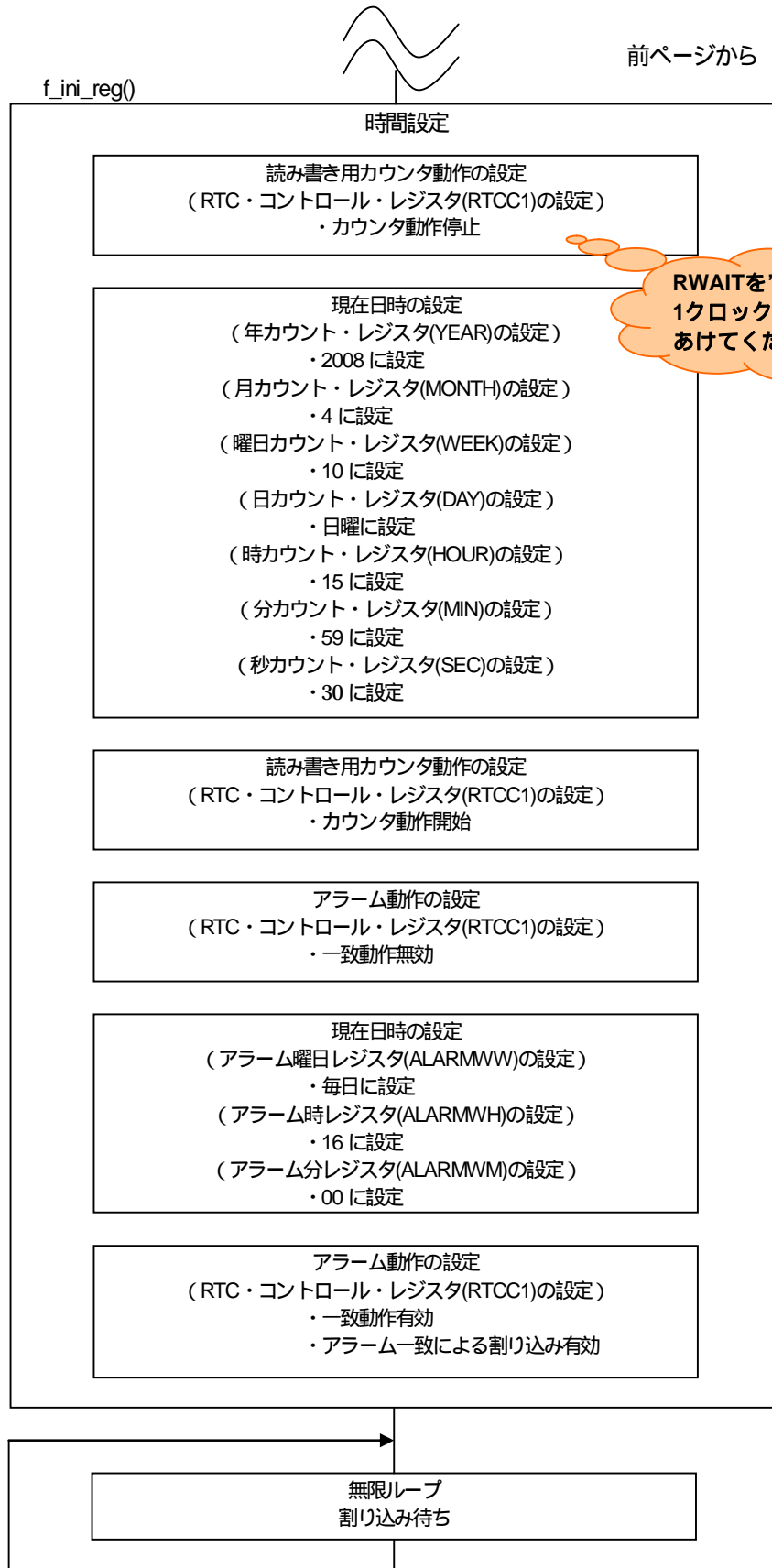


### 3.4 フロー・チャート

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。

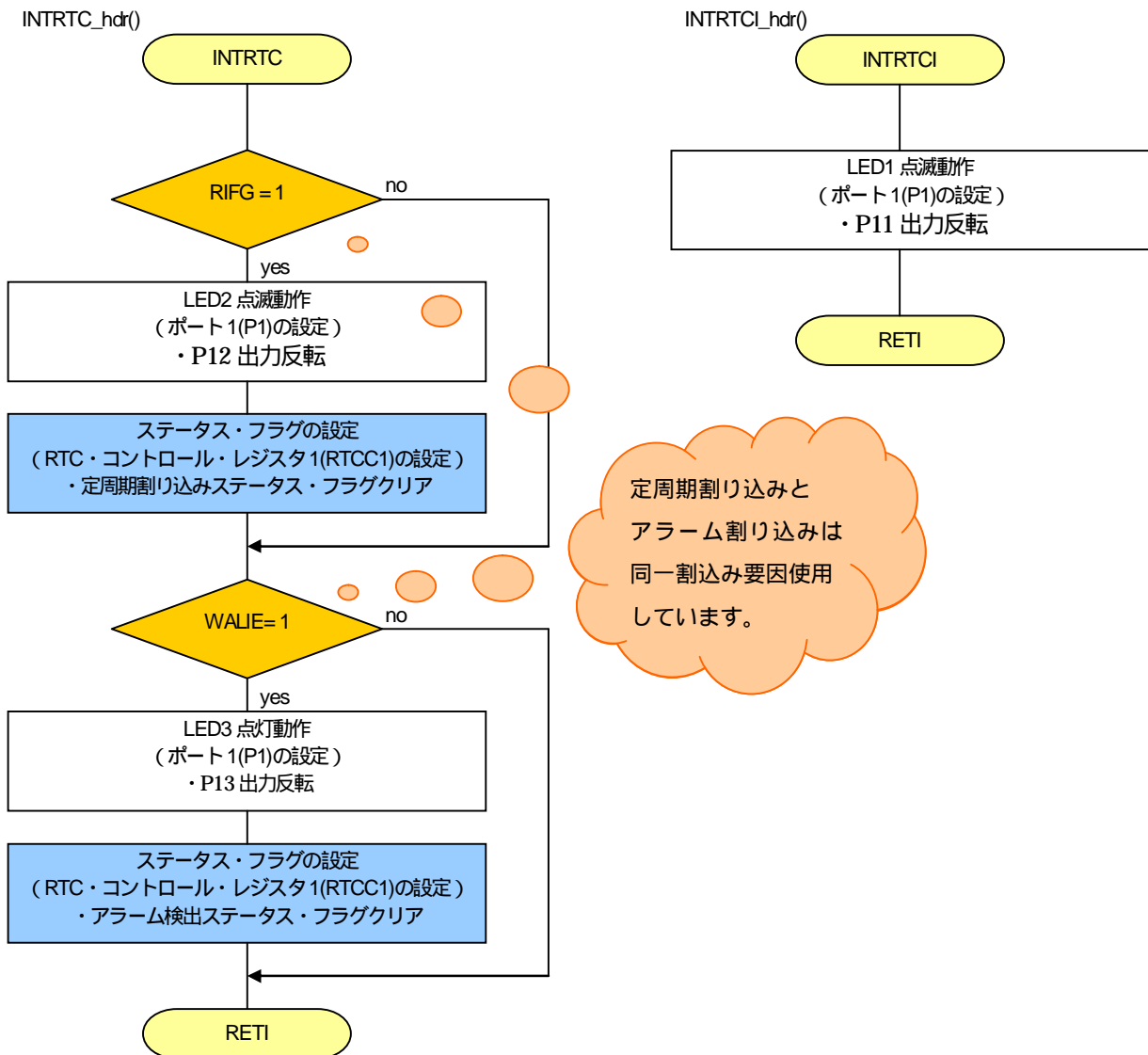
(1) 全体図フロー





- 注1. オプション・バイトについては，サンプル・プログラム初期設定編アプリケーション・ノート  
の“4.6 オプション・バイトの設定概要”を参照して下さい。
2. 入出力ポートの設定 (f\_ini\_port()) の設定については，サンプル・プログラム初期設定編ア  
プリケーション・ノートの“3.4 フロー・チャート”を参照して下さい。
3. 低電圧検出設定 (f\_ini\_lvi()) については，サンプル・プログラム低電圧検出回路編アプリケー  
ション・ノートの“3.4 フロー・チャート”を参照して下さい。
4. CPU初期設定 (f\_ini\_cpu()) については，サンプル・プログラム初期設定編アプリケーション・  
ノートの“3.4 フロー・チャート”を参照して下さい。

(2) 割り込み処理フロー



## 第4章 設定方法について

この章では、リアルタイム・カウンタを使用して、定周期割り込み、インターバル割り込み、およびアラーム割り込みを発生させる場合の設定について説明します。

その他の初期設定については、78K0R/Kx3-L サンプル・プログラム(初期設定編)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、各製品のユーザズ・マニュアル([78K0R/Kx3-L](#))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 ユーザズ・マニュアル](#)を参照してください。

### 4.1 リアルタイム・カウンタの設定

リアルタイム・カウンタを使用する場合、主に次のレジスタの設定、操作が必要になります。

#### 初期設定で使用するレジスタ

- ・周辺イネーブル・レジスタ0 (PER0)
- ・リアルタイム・カウンタ・コントロール・レジスタ0 (RTCC0)
- ・リアルタイム・カウンタ・コントロール・レジスタ1 (RTCC1)
- ・リアルタイム・カウンタ・コントロール・レジスタ2 (RTCC2)
- ・秒カウント・レジスタ (SEC)
- ・分カウント・レジスタ (MIN)
- ・時カウント・レジスタ (HOUR)
- ・日カウント・レジスタ (DAY)
- ・曜日カウント・レジスタ (WEEK)
- ・月カウント・レジスタ (MONTH)
- ・年カウント・レジスタ (YEAR)
- ・アラーム分レジスタ (ALARMWWM)
- ・アラーム時レジスタ (ALARMWH)
- ・アラーム曜日レジスタ (ALARMWW)

### 4.2 割り込みの設定

#### 割り込み操作で使用するレジスタ

- ・割り込み要求フラグ・レジスタ(RTCIF)
- ・割り込み要求フラグ・レジスタ(RTCIIF)
- ・割り込みマスク・フラグ・レジスタ(RTCMK)
- ・割り込みマスク・フラグ・レジスタ(RTCIMK)

### 4.3 リアルタイム・カウンタの設定レジスタ

(1) 周辺イネーブル・レジスタ0 (PER0)

PER0は、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。

リアルタイム・カウンタを使用するときは、必ずビット7 (RTCEN) を1に設定してください。

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。

リセット信号の発生により、00Hになります。

略号： PER0

7	6	5	4	3	2	1	0
RTCEN	0	ADCEN	0	IICAEN	SAU0EN	0	0

RTCEN	リアルタイム・カウンタ (RTC) の入力クロックの制御
0	入力クロック供給停止 ・リアルタイム・カウンタ (RTC) で使用するSFRへのライト不可 (リード可) ・リアルタイム・カウンタ (RTC) の動作は継続可能
1	入力クロック供給許可 ・リアルタイム・カウンタ (RTC) で使用するSFRへのリード/ライト可

ADCEN	A/Dコンバータの入力クロックの制御
0	入力クロック供給停止 ・A/Dコンバータで使用するSFRへのライト不可 ・A/Dコンバータはリセット状態
1	入力クロック供給許可 ・A/Dコンバータで使用するSFRへのリード/ライト可

IICAEN	シリアル・インターフェースIICAの入力クロックの制御
0	入力クロック供給停止 ・シリアル・インターフェースIICAで使用するSFRへのライト不可 ・シリアル・インターフェースIICAはリセット状態
1	入力クロック供給許可 ・シリアル・インターフェースIICAで使用するSFRへのリード/ライト可

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

注：リアルタイム・カウンタを使用する場合は、サブシステム・クロック (fSUB) が発振安定状態において、最初にRTCEN = 1の設定を行ってください。RTCEN = 0の場合は、リアルタイム・カウンタの制御レジスタへの書き込みは無視され、読み出し値もすべて初期値となります。

(2) リアルタイム・カウンタ・コントロール・レジスタ0 (RTCC0)

RTCC0は、リアルタイム・カウンタ動作の開始/停止、RTCCL端子/RTC1HZ端子の制御、12/24時間制、定周期割り込み機能を設定するレジスタです。

RTCC0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。

リセット信号の発生により、00Hになります。

略号： RTCC0

	7	6	5	4	3	2	1	0
RTCC0	RTCE	0	RCLOE1	RCLOE0	AMPM	CT2	CT1	CT0

RTCE	リアルタイム・カウンタの動作制御
0	カウンタ動作停止
1	カウンタ動作開始

RTCLOE1	RTC1HZ端子の出力制御
0	RTC1HZ端子の出力 (1Hz) 禁止
1	RTC1HZ端子の出力 (1Hz) 許可

RTCLOE0	RTCCL端子の出力制御
0	RTCCL端子の出力 (32kHz) 禁止
1	RTCCL端子の出力 (32kHz) 許可

AMPM	12時間制 / 24時間制の選択
0	12時間制 (午前 / 午後を表示)
1	24時間制

CT2	CT1	CT0	定周期割り込み (INTRTC) の選択
0	0	0	定周期割り込み機能を使用しない
0	0	1	0.5秒に1度 (秒カウントアップに同期)
0	1	0	1秒に1度 (秒カウントアップと同時)
0	1	1	1分に1度 (毎分00秒)
1	0	0	1時間に1度 (毎時00分00秒)
1	0	1	1日に1度 (毎日00時00分00秒)
1	1	x	1月に1度 (毎月1日午前00時00分00秒)

(3) リアルタイム・カウンタ・コントロール・レジスタ1 (RTCC1)

アラーム割り込み機能，カウンタのウェイトを制御するレジスタです。

RTCC0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。

リセット信号の発生により，00Hになります。

略号： RTCC1

7	6	5	4	3	2	1	0
WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT

WALE	アラームの動作制御
0	一致動作無効
1	一致動作有効

WALIE	アラーム割り込み (INTRTC) 機能の動作制御
0	アラームの一致による割り込みを発生しない
1	アラームの一致による割り込みを発生する

WAFG	アラーム検出ステータス・フラグ
0	アラーム不一致
1	アラームの一致検出

RIFG	定周期割り込みステータス・フラグ
0	定周期割り込み発生なし
1	定周期割り込み発生あり

RWST	リアルタイム・カウンタのウェイト状態フラグ
0	カウンタ動作中
1	カウンタ値の読み出し，書き込みモード中

RWAIT	リアルタイム・カウンタのウェイト制御
0	カウンタ動作設定
1	SEC～YEARカウンタ停止設定。カウンタ値読み出し，書き込みモード

**注意** RTCC1レジスタに1ビット操作命令で書き込みを行うと，RIFGフラグ，WAFGフラグがクリアされることがあります。そのため，RTCC1レジスタへの書き込みは8ビット操作命令で設定してください。書き込み時に，RIFGフラグ，WAFGフラグをクリアしないようにするためには，該当ビットに書き込みが無効となる“1”を設定してください。

**備考** 定周期割り込みとアラーム一致割り込みは，同一割り込み要因 (INTRTC) を使用しています。定周期割り込みステータス・フラグ (RIFG) とアラーム検出ステータス・フラグ (WAFG) を確認することで，どちらの割り込みが発生したかを判断することができます。



(4) リアルタイム・カウンタ・コントロール・レジスタ2 (RTCC2)

インターバル割り込み機能, RTCDIV端子を制御する8ビットのレジスタです。  
RTCC2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。  
リセット信号の発生により, 00Hになります。

略号: RTCC2

7	6	5	4	3	2	1	0
RINTE	RCLOE2	RCKDIV	0	0	ICT2	ICT1	ICT0

RINTE	ICT2	ICT1	ICT0	インターバル割り込み (INTRTCI) の選択
0	x	x	x	インターバル割り込みを発生しない
1	0	0	0	$2^6/f_{XT}$ ( 1.953125ms )
1	0	0	1	$2^7/f_{XT}$ ( 3.90625ms )
1	0	1	0	$2^8/f_{XT}$ ( 7.8125ms )
1	0	1	1	$2^9/f_{XT}$ ( 15.625ms )
1	1	0	0	$2^{10}/f_{XT}$ ( 31.25ms )
1	1	0	1	$2^{11}/f_{XT}$ ( 62.5ms )
1	1	1	x	$2^{12}/f_{XT}$ ( 125ms )

RCLOE2	RTCDIV端子の出力制御
0	RTCDIV端子の出力禁止
1	RTCDIV端子の出力許可

RCKDIV	RTCDIV端子の出力周波数の選択
0	RTCDIV端子から512 Hzを出力
1	RTCDIV端子から16.384 kHzを出力

(5) 年, 月, 日, 曜, 時, 分, 秒カウント・レジスタ (YEAR, MONTH, DAY, WEEK, HOUR, MIN, SEC)

略号: YEAR

7	6	5	4	3	2	1	0
YEAR80	YEAR40	YEAR20	YEAR10	YEAR8	YEAR4	YEAR2	YEAR1

0-99 (10進) までの値を取り, 年のカウント値をBCDコードで示す8ビットのレジスタです。  
00, 04, 08, ..., 92, 96がうるう年となります。

略号: MONTH

7	6	5	4	3	2	1	0
0	0	0	MONTH10	MONTH 8	MONTH4	MONTH2	MONTH1

MONTHレジスタは1-12 (10進) までの値を取り, 月のカウント値をBCDコードで示す8ビットのレジスタです。

略号: DAY

7	6	5	4	3	2	1	0
0	0	DAY20	DAY10	DAY8	DAY4	DAY2	DAY1

1-31 (10進) までの値を取り, 日のカウント値をBCDコードで示す8ビットのレジスタです。

略号: WEEK

7	6	5	4	3	2	1	0
0	0	0	0	0	WEEK4	WEEK2	WEEK1

0-6 (10進) までの値を取り, 曜日のカウント値を示す8ビットのレジスタです。

略号: HOUR

7	6	5	4	3	2	1	0
0	0	HORU20	HORU10	HORU8	HORU4	HORU2	HORU1

0-23または1-12 (10進) までの値を取り, 時のカウント値をBCDコードで示す8ビットのレジスタです。

略号: MIN

7	6	5	4	3	2	1	0
0	MIN40	MIN20	MIN10	MIN8	MIN4	MIN2	MIN1

0-59 (10進) までの値を取り, 分のカウント値をBCDコードで示す8ビットのレジスタです。

略号: SEC

7	6	5	4	3	2	1	0
0	SEC40	SEC20	SEC10	SEC8	SEC4	SEC2	SEC1

0-59 (10進) までの値を取り, 秒のカウント値をBCDコードで示す8ビットのレジスタです。

(6) アラーム曜，時，分レジスタ (ALARMWW，ALARMWH，ALARMWM)

略号： ALARMWW

7	6	5	4	3	2	1	0
0	WW6	WW5	WW4	WW3	WW2	WW1	WW0

アラームの曜日を設定するレジスタです。

略号：ALARMWH

7	6	5	4	3	2	1	0
0	0	WH20	WH10	WH8	WH4	WH2	WH1

アラームの時をBCDコードで設定するレジスタです。

略号：ALARMWM

7	6	5	4	3	2	1	0
0	WM40	WM20	WM10	WM8	WM4	WM2	WM1

アラームの分をBCDコードで設定するレジスタです。

下表はアラーム曜 (ALARMWW) の設定例です。

	7	6	5	4	3	2	1	0
アラーム	0	WW6 (土)	WW5 (金)	WW4 (木)	WW3 (水)	WW2 (火)	WW1 (月)	WW0 (日)
毎日	0	1	1	1	1	1	1	1
月,水,金	0	0	1	0	1	0	1	0
日	0	0	0	0	0	0	0	1
金,土	0	1	1	0	0	0	0	0

## 4.4 割り込みの設定レジスタ

### (1) 割り込み要求フラグ・レジスタ(IF1H)

割り込み要求フラグは、対応する割り込み要求の発生または命令の実行によりセット（1）され、割り込み要求受け付け時、リセット信号発生時または命令の実行によりクリア（0）されるフラグです。

割り込みが受け付けられた場合、まず割り込み要求フラグが自動的にクリアされてから割り込みルーチンに入ります。

IF1H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。

また、IF1LとIF1Hをあわせて16ビット・レジスタIF1として使用するときには、16ビット・メモリ操作命令で設定します。

リセット信号の発生により、00Hになります。

略号：IF1H

7	6	5	4	3	2	1	0
TMIF04	MDIF	0	0	KRIF	RTCIIF	RTCIF	ADIF

xxIFxx	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

注意1．IF1Hのビット4、5には必ず0を設定してください。

2．タイマ、シリアル・インターフェース、A/Dコンバータなどをスタンバイ解除後に動作させる場合、

いったん割り込み要求フラグをクリアしてから動作させてください。ノイズなどにより割り込み要求フラグがセットされる場合があります。

3．割り込み要求フラグ・レジスタのフラグ操作には、1ビット・メモリ操作命令（CLR1）を使用してください。C言語での記述の場合は、コンパイルされたアセンブラが1ビット・メモリ操作命令（CLR1）になっている必要があるため、「IF0L.0 = 0;」や「\_asm("clr1 IF0L,0");」のようなビット操作命令を使用してください。

なお、C言語で「IF0L & = 0xfe;」のように8ビット・メモリ操作命令で記述した場合、コンパイルすると3命令のアセンブラになります。

```
MOV  A    ,   IF0L
AND  A    ,   #0FEH
MOV  IF0L ,   A
```

この場合、「MOV A, IF0L」後から「MOV IF0L, A」の間のタイミングで、同一の割り込み要求フラグ・レジスタ（IF0L）の他ビットの要求フラグがセット（1）されても、「MOV IF0L, A」でクリア（0）されます。

したがって、C言語で8ビット・メモリ操作命令を使用する場合は注意が必要です。

(2) 割り込み要求フラグ・レジスタ(MK1H)

割り込みマスク・フラグは、対応するマスカブル割り込み処理の許可 / 禁止を設定するフラグです。

MK1H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。また、MK1LとMK1Hをあわせて16ビット・レジスタMK1として使用するときには、16ビット・メモリ操作命令で設定します。

リセット信号の発生により、FFHになります。

**備考** このレジスタへの書き込み命令を行った場合、命令実行クロック数が2クロック長くなります。

略号 : MK1H

7	6	5	4	3	2	1	0
TMMK04	MDMK	0	0	KRMK	RTCIMK	RTCMK	ADMK

xxMKxx	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

## 4.5 時計誤差補正について

リアルタイム・カウンタの時計誤差補正について説明します。

### (1) 時計誤差補正レジスタ (SUBCUD)

リアルタイム・カウンタの時計誤差補正を行う場合、時計誤差補正レジスタ (SUBCUD) を使用します。時計誤差補正レジスタ (SUBCUD) は、サブカウント・レジスタ (RSUBC) から秒カウンタ・レジスタへオーバーフローする値 (基準値: 7FFFH) を変化させることにより、時計の進みや遅れをより高精度に補正することができるレジスタです。

図4 - 1 時計誤差補正レジスタ (SUBCUD) のフォーマット

DEV	F6	F5	F4	F3	F2	F1	F0						
		<table border="1"> <thead> <tr> <th>F6</th> <th>時計誤差補正値の設定</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>{ (F5, F4, F3, F2, F1, F0) - 1 } × 2だけ増加</td> </tr> <tr> <td>1</td> <td>{ (/F5, /F4, /F3, /F2, /F1, /F0) + 1 } × 2だけ減少<sup>注</sup></td> </tr> </tbody> </table>						F6	時計誤差補正値の設定	0	{ (F5, F4, F3, F2, F1, F0) - 1 } × 2だけ増加	1	{ (/F5, /F4, /F3, /F2, /F1, /F0) + 1 } × 2だけ減少 <sup>注</sup>
F6	時計誤差補正値の設定												
0	{ (F5, F4, F3, F2, F1, F0) - 1 } × 2だけ増加												
1	{ (/F5, /F4, /F3, /F2, /F1, /F0) + 1 } × 2だけ減少 <sup>注</sup>												
		<table border="1"> <thead> <tr> <th>DEV</th> <th>時計誤差補正のタイミングの設定</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>秒桁が00, 20, 40時 (20秒ごと) に時計誤差補正</td> </tr> <tr> <td>1</td> <td>秒桁が00時のみ (60秒ごと) に時計誤差補正</td> </tr> </tbody> </table>						DEV	時計誤差補正のタイミングの設定	0	秒桁が00, 20, 40時 (20秒ごと) に時計誤差補正	1	秒桁が00時のみ (60秒ごと) に時計誤差補正
DEV	時計誤差補正のタイミングの設定												
0	秒桁が00, 20, 40時 (20秒ごと) に時計誤差補正												
1	秒桁が00時のみ (60秒ごと) に時計誤差補正												

注. /F5~/F0は、ビット反転した値 (111100のときは000011) となります。

注意1. (F6, F5, F4, F3, F2, F1, F0) = (\*, 0, 0, 0, 0, 0, \*) のときは、時計誤差補正を行いません (\* は0または1)。

補正値の範囲: (F6=0のとき) 2, 4, 6, 8, ……120, 122, 124

(F6=1のとき) -2, -4, -6, -8, ……-120, -122, -124

2. 時計誤差補正レジスタ (SUBCUD) による補正可能範囲は次のとおりです。

	DEV = 0 (20秒ごとの補正)	DEV = 1 (60秒ごとの補正)
補正可能範囲	- 189.2 ppm ~ 189.2 ppm	- 63.1 ppm ~ 63.1 ppm
最大量子化誤差	± 1.53 ppm	± 0.51 ppm
最小分解能	± 3.05 ppm	± 1.02 ppm

3. 時計誤差補正を行う場合にF6-F0に設定する補正値 (2, 4, 6, ……124, または -2, -4, -6, ……-124) は、次の式で算出できます。

(DEV = 0の場合)

補正値 = 1分間の補正カウント数 ÷ 3

= (発振周波数 ÷ ターゲット周波数 - 1) × 32768 × 60 ÷ 3

(DEV = 1の場合)

補正値 = 1分間の補正カウント数

= (発振周波数 ÷ ターゲット周波数 - 1) × 32768 × 60

発振周波数 : サブシステム・クロック (f<sub>SUB</sub>) の値

ターゲット周波数 : 時計誤差補正レジスタを使用した補正後の周波数

備考. 補正範囲が, - 63.1 ppm以下または63.1 ppm以上のときは, DEV = 0を設定してください。

(2) 時計誤差補正例

リアルタイム・カウンタの時計誤差補正例を次に示します。

【32772.3 Hz ( 32768 Hz + 131.2 ppm ) の場合の補正例】

RTCCL端子またはRTC1HZ端子の出力<sup>注1</sup>を使用し、発振周波数 ( 32772.3 Hz ) をマイコン外部に出力します。

マイコン外部で、出力した発振周波数を周波数カウンタなどの計測器で測定し、ターゲット周波数 ( 32768 Hz ) との差分から補正値を算出します<sup>注2</sup>。

補正値をシリアル・インターフェースなどで受信し、時計誤差補正レジスタ ( SUBCUD ) に設定します<sup>注3</sup>。

注1 . RTC1Hz端子およびRTCCL端子の出力の設定手順については、ユーザズ・マニュアルを参照してください。

2 . この場合、20秒ごとの補正が必要となるため、補正値の算出式は次のようになります。

$$\begin{aligned}
 \text{補正値} &= \text{1分間の補正カウント数} \div 3 \\
 &= (\text{発振周波数の1分間のカウント数} - \text{ターゲット周波数の1分間のカウント数}) \div 3 \\
 &= (\text{発振周波数} \div \text{ターゲット周波数} \times 32768 \times 60 - 32768 \times 60) \div 3 \\
 &= (\text{発振周波数} \div \text{ターゲット周波数} - 1) \times 32768 \times 60 \div 3 \\
 &= (32772.3 \div 32768 - 1) \times 32768 \times 60 \div 3 \\
 &= 86
 \end{aligned}$$

3 . 時計誤差補正レジスタ ( SUBCUD ) への設定値は次のとおりです

・ビット7 ( DEV )

20秒ごとの補正が必要となるため、DEV = 0とします。

・ビット6-0 ( F6-F0 )

補正値が0以上 ( 遅くする場合 ) では、F6 = 0とします。

( F5 , F4 , F3 , F2 , F1 , F0 ) は、補正値から算出します。

$$\{ ( F5 , F4 , F3 , F2 , F1 , F0 ) - 1 \} \times 2 = 86$$

$$( F5 , F4 , F3 , F2 , F1 , F0 ) = 44$$

$$( F5 , F4 , F3 , F2 , F1 , F0 ) = ( 1 , 0 , 1 , 1 , 0 , 0 )$$

## 4.6 RTC設定の初期設定概要

RTCを動作させ、定周期割り込み、インターバル割り込みを発生させる場合は、次の手順で実行します。

以降のソースコードの例は、本サンプル・プログラム・コードと同内容としています。

### RTC へのクロック供給開始

周辺イネーブル・レジスタ0 (PER0)  
RTCへのクロック供給を開始

略号 : PER0

	7	6	5	4	3	2	1	0
RTCEN	0	ADCEN	0	IICAEN	SAU0EN	0	0	0
	1	0	x	0	x	x	0	0

RTCEN	RTCの入力クロックの制御
0	入力クロック供給停止
1	入力クロック供給

#### 【使用例】

RTCハードウェア・マクロを**使用可能**とする場合

アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SET1    RTCEN
```

C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
RTCEN = 1;
```



定周期割り込みの設定

RTC・コントロール・レジスタ0 (RTCC0)  
時刻表現設定  
定周期割り込みの周期設定

略号：RTCC0

7	6	5	4	3	2	1	0
RTCE	0	RCLOE1	RCLOE0	AMPM	CT2	CT1	CT0
0	0	0	0	1	0	0	1

CT2	CT1	CT0	定周期割り込み (INTRTC) の選択
0	0	0	定周期割り込み機能を使用しない
0	0	1	0.5秒に1度 (秒カウントアップに同期)
0	1	0	1秒に1度 (秒カウントアップと同時)
0	1	1	1分に1度 (毎分00秒)
1	0	0	1時間に1度 (毎時00分00秒)
1	0	1	1日に1度 (毎日00時00分00秒)
1	1	x	1月に1度 (毎月1日午前00時00分00秒)

AMPM	12時間制 / 24時間制の選択
0	12時間制 (午前 / 午後を表示)
1	24時間制

ここでは、  
RTCEは0のまま  
とします。

RTCE	リアルタイム・カウンタの動作制御
0	カウンタ動作停止
1	カウンタ動作開始

【使用例】

1秒周期での定周期割り込み使用し、24時間制表現でのRTC設定とする場合

```
アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)
MOV RTCC0, #09H
```

```
C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)
RTCC0 = 0b00001001;
```

インターバル割り込み設定

RTC・コントロール・レジスタ2 (RTCC2)  
インターバル割り込みの周期設定

略号 : RTCC2

7	6	5	4	3	2	1	0
RINTE	RCLOE2	RCKDIV	0	0	ICT2	ICT1	ICT0
1	0	0	0	0	1	1	x

RINTE	ICT2	ICT1	ICT0	インターバル割り込み (INTRTC1) の選択
0	x	x	x	インターバル割り込みを発生しない
1	0	0	0	$2^6 / f_{XT}$ (1.953125ms)
1	0	0	1	$2^7 / f_{XT}$ (3.90625ms)
1	0	1	0	$2^8 / f_{XT}$ (7.8125ms)
1	0	1	1	$2^9 / f_{XT}$ (15.625ms)
1	1	0	0	$2^{10} / f_{XT}$ (31.25ms)
1	1	0	1	$2^{11} / f_{XT}$ (62.5ms)
1	1	1	x	$2^{12} / f_{XT}$ (125ms)

【使用例】

125ms周期でのインターバル割り込みを使用する場合

アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOV RTCC2, #87H
```

C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
RTCC2 = 0b10000111;
```

年月日曜時分秒の設定 (初期設定)

年月日曜時分秒カウント・レジスタ  
(YEAR, MONTH, DAY, WEEK, HOUR, MIN, SEC)  
日時を設定

略号 : YEAR

7	6	5	4	3	2	1	0
00 - 99 ( BCDコード )							

略号 : MONTH

7	6	5	4	3	2	1	0	
0	0	0	01 - 12 ( BCDコード )					

略号 : DAY

7	6	5	4	3	2	1	0
0	0	0 - 31 ( BCDコード )					

略号 : WEEK

7	6	5	4	3	2	1	0
0	0	0	0	0	0 - 6		

略号 : HOUR

7	6	5	4	3	2	1	0
0	0	0 - 23 または 1 - 12 , 21 - 32 ( BCDコード )					

略号 : MIN

7	6	5	4	3	2	1	0
0	0 - 59 ( BCDコード )						

略号 : SEC

7	6	5	4	3	2	1	0
0	0 - 59 ( BCDコード )						

【使用例】

08 / 4 / 10 ( SUN ) 15 : 59 : 30 の設定を行う場合

アセンブリ言語の場合 ( 78K0R/KE3-Lマイクロコントローラ使用時 )

```

MOV     SEC      , #30H
MOV     MIN      , #59H
MOV     HOUR     , #15H
MOV     WEEK     , #00H
MOV     DAY      , #10H
MOV     MONTH    , #04H
MOV     YEAR     , #08H
    
```

C言語の場合 ( 78K0R/KE3-Lマイクロコントローラ使用時 )

```

SEC      = 0x30;
MIN      = 0x59;
HOUR     = 0x15;
WEEK     = 0x00;
DAY      = 0x10;
MONTH    = 0x04;
YEAR     = 0x08;
    
```

割り込み設定

割り込み要求フラグ・レジスタ (IF1H)  
 RTCIIF, RTCIF割り込み要因フラグをクリア  
 割り込みマスク・フラグ・レジスタ (MK1H)  
 RTCIMK, RTCMK割り込みマスク解除

略号 : IF1H

7	6	5	4	3	2	1	0
TMIF04	MDIF	0	0	KRIF	RTCIIF	RTCIF	ADIF
x	x	0	0	x	0	0	x

xxIF	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

略号 : MK1H

7	6	5	4	3	2	1	0
TMMK04	MDMK	0	0	KRMK	RTCIMK	RTCMK	ADMK
x	x	0	0	x	0	0	x

xxMK	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

【使用例】

定周期割り込み、インターバル割り込みを許可する場合

アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
CLR1    RTCIF
CLR1    RTCIIF
CLR1    RTCMK
CLR1    RTCIMK
```

C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
RTCIF    = 0;
RTCIIF   = 0;
RTCMK    = 0;
RTCIMK   = 0;
```

RTC の動作開始

RTC・コントロール・レジスタ0 (RTCC0)  
カウント動作を開始

略号 : RTCC0

7	6	5	4	3	2	1	0
RTCE	0	RCLOE1	RCLOE0	AMPM	CT2	CT1	CT0
1	0	x	x	x	x		

RTCE	リアルタイム・カウンタの動作制御
0	カウンタ動作停止
1	カウンタ動作開始

【使用例】

RTC動作を開始する場合

アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

SET1      RTCE

C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

RTCE = 1;

## 4.7 RTC読み出し / 書き込みを行う際の設定概要

動作中のRTCのカウンタの読み出し / 書き込みは、次の手順で実行します。

### RTC の読み出し / 書き込み開始

リアルタイム・カウンタ・コントロール・レジスタ1 (RTCC1)  
RWAITによる停止と、RWSTによる停止確認

略号 : RTCC1

7	6	5	4	3	2	1	0
WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT
x	x	0	x	x	0	1	1

RWAIT	リアルタイム・カウンタのウェイト制御
0	カウンタ動作設定
1	SEC~YEARカウンタ停止設定。カウンタ値読み出し、書き込みモード

RWST	リアルタイム・カウンタのウェイト状態フラグ
0	カウンタ動作中
1	カウンタ値の読み出し、書き込みモード中

### RTC の読み出し / 書き込み

年月日曜時分秒カウント・レジスタ  
(YEAR, MONTH, DAY, WEEK, HOUR, MIN, SEC)  
カウント値の読み出しまたは書き込み

RTCの読み出し/書き込み終了

RTC・コントロール・レジスタ1 (RTCC1)  
RWAITによる開始と、RWSTによる開始確認

略号：RTCC1

	7	6	5	4	3	2	1	0
WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT	
x	x	0	x	x	0	0	0	

RWAIT	リアルタイム・カウンタのウェイト制御
0	カウンタ動作設定
1	SEC~YEARカウンタ停止設定。カウンタ値読み出し、書き込みモード

RWST	リアルタイム・カウンタのウェイト状態フラグ
0	カウンタ動作中
1	カウンタ値の読み出し、書き込みモード中

【使用例】

RTCカウンタ値の読み出し、または書き込みを行う場合

アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```

SET1   RWAIT
APP_1:
BT     RWST      ,   $APP_2
BR     $APP_1
APP_2:
;カウンタの読み出しまたは書き込みを行います
APP_3:
BT     RWST      ,   $APP_3
    
```

C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```

RWAIT = 1;
while ( !RWST );
/* カウンタの読み出しまたは書き込みを行います */
RWAIT = 0;
while ( RWST );
    
```

## 4.8 RTCアラームを使用する際の設定概要

RTCのアラーム設定は、次の手順で実行します。

### RTC のアラーム設定開始

RTC・コントロール・レジスタ1 (RTCC1)  
 アラームの一致動作を無効  
 アラームの一致割り込みを有効

略号：RTCC1

7	6	5	4	3	2	1	0
WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT
0	1	0	x	x	0	x	x

WALIE	アラーム割り込み (INTRTC) 機能の動作制御
0	アラームの一致による割り込みを発生しない
1	アラームの一致による割り込みを発生する

WALE	アラームの動作制御
0	一致動作無効
1	一致動作有効

### RTC のアラーム設定開始

アラーム曜、時、分レジスタ (ALARMWW, ALARMWH, ALARMWM)  
 アラーム時間の設定

略号：ALARMWW

7	6	5	4	3	2	1	0
0	WW6	WW5	WW4	WW3	WW2	WW1	WW0
0	アラーム曜 1 6						

略号：ALARMWH

7	6	5	4	3	2	1	0
0	0	WH20	WH10	WH8	WH4	WH2	WH1
0	0	アラーム時 0 - 23 または 1 - 12, 21 - 32 (BCDコード)					

略号：ALARMWM

7	6	5	4	3	2	1	0
0	WM40	WM20	WM10	WM8	WM4	WM2	WM1
0	アラーム分 0 - 59 (BCDコード)						



RTCのアラーム設定終了

リアルタイム・カウンタ・コントロール・レジスタ1 (RTCC1)  
アラームの一致動作を有効

略号 : RTCC1

	7	6	5	4	3	2	1	0
	WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT
	1	1	0	x	x	0	x	x

WALE	アラームの動作制御
0	一致動作無効
1	一致動作有効

【使用例】

毎日16時00分にアラーム動作を行う場合

アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
CLR1    WALE
SET1    WALIE
MOV     ALARMWMM , #00H
MOV     ALARMWH  , #16H
MOV     ALARMWW  , #7FH
SET1    WALE
```

C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
WALE    = 0;
WALIE   = 1;
ALARMWMM = 0;
ALARMWH  = 0x16;
ALARMWW  = 0x7F;
WALE    = 1;
```





## 第5章 PM+を用いたHEXファイルの生成

この章では、PM+とダウンロードしたC言語用のファイルを用い、サンプル・プログラムからHEXファイルに生成する方法を説明します。





### 5.1 ダウンロードファイルの解説

ダウンロードした各種ファイルの説明をします。


#### (1) C言語版

	ファイル名	内容
	78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル
	78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ
	Kx3-L_RTC.c	リアルタイム・カウンタのC言語ソース・ファイル
	OP.asm	オプション・バイトのアセンブリ言語ソース・ファイル

#### (2) アセンブラ版

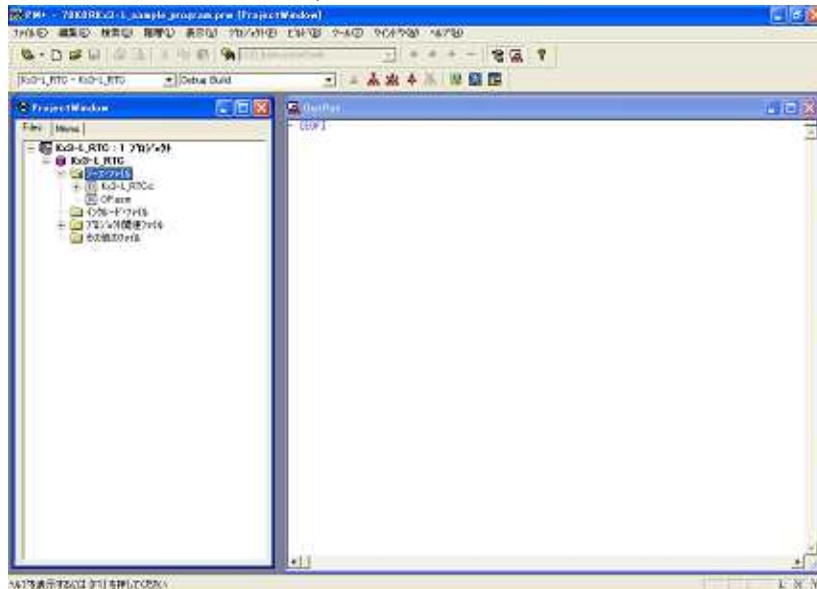
	ファイル名	内容
	78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル
	78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ
	Kx3-L_RTC.asm	リアルタイム・カウンタのC言語ソース・ファイル
	OP.asm	オプション・バイトのアセンブリ言語ソース・ファイル

## 5.2 サンプル・プログラムのHEXファイル生成（ビルド）

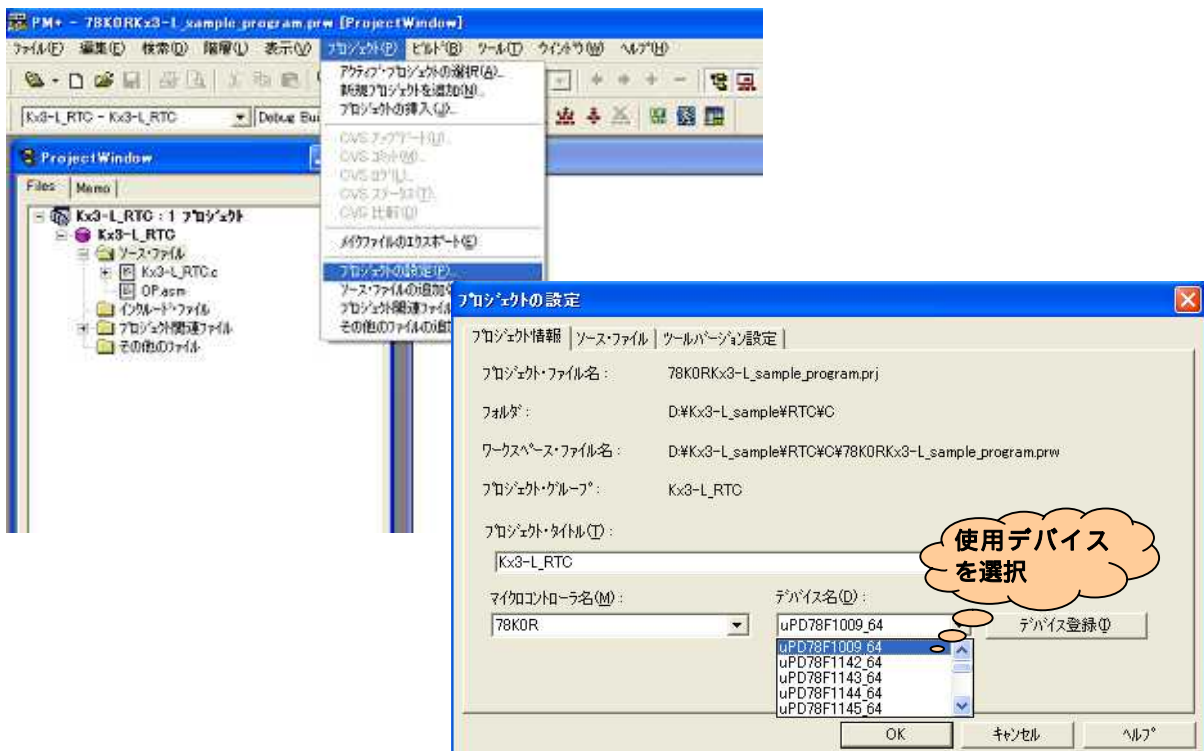
サンプル・プログラムからHEXファイルに生成するために、PM+を用いてサンプル・プログラムをビルドする必要があります。ここでは、でダウンロードしたC言語版のファイルを用いて、統合開発環境PM+にてビルドしてから、HEXファイルを生成するまでの動作の一例を説明します。

PM+操作方法の詳細については、[PM+ プロジェクト・マネージャ ユーザーズ・マニュアル](#)を参照してください。

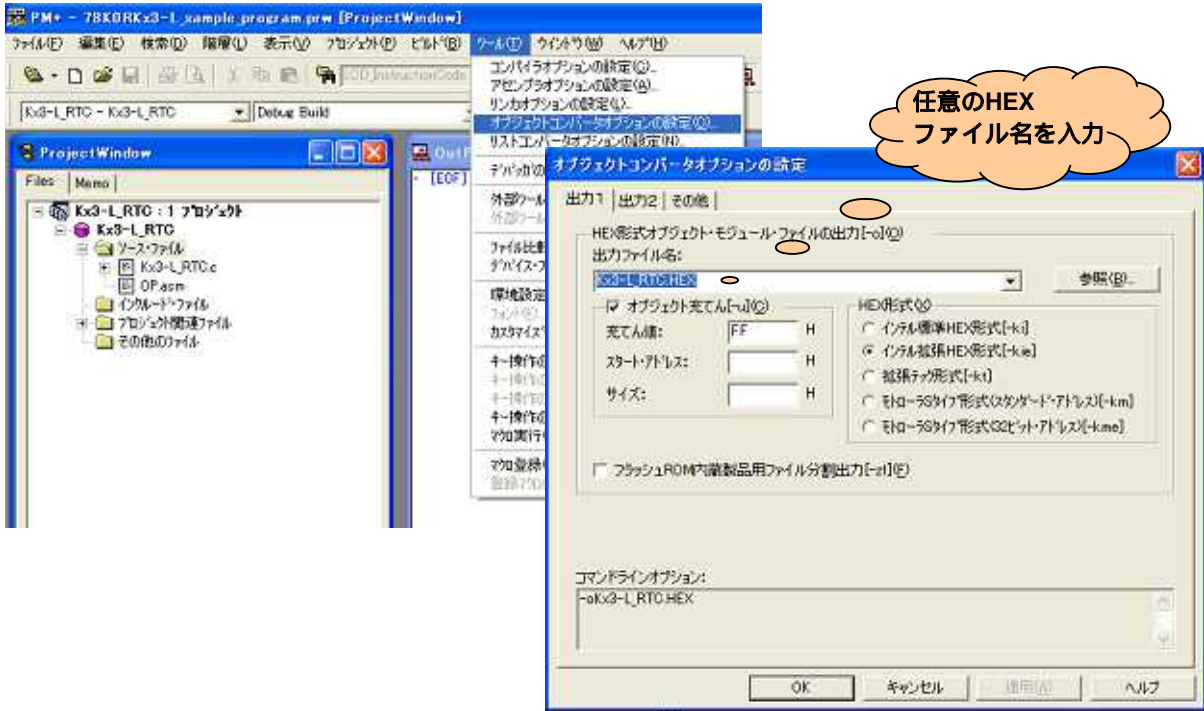
- (1) ダウンロードしたファイルを解凍し、「78K0RKx3-L\_sample\_program.prw」をダブルクリックしてください。ワークスペースが開き、その中にソース・ファイルが自動的に読み込まれます。




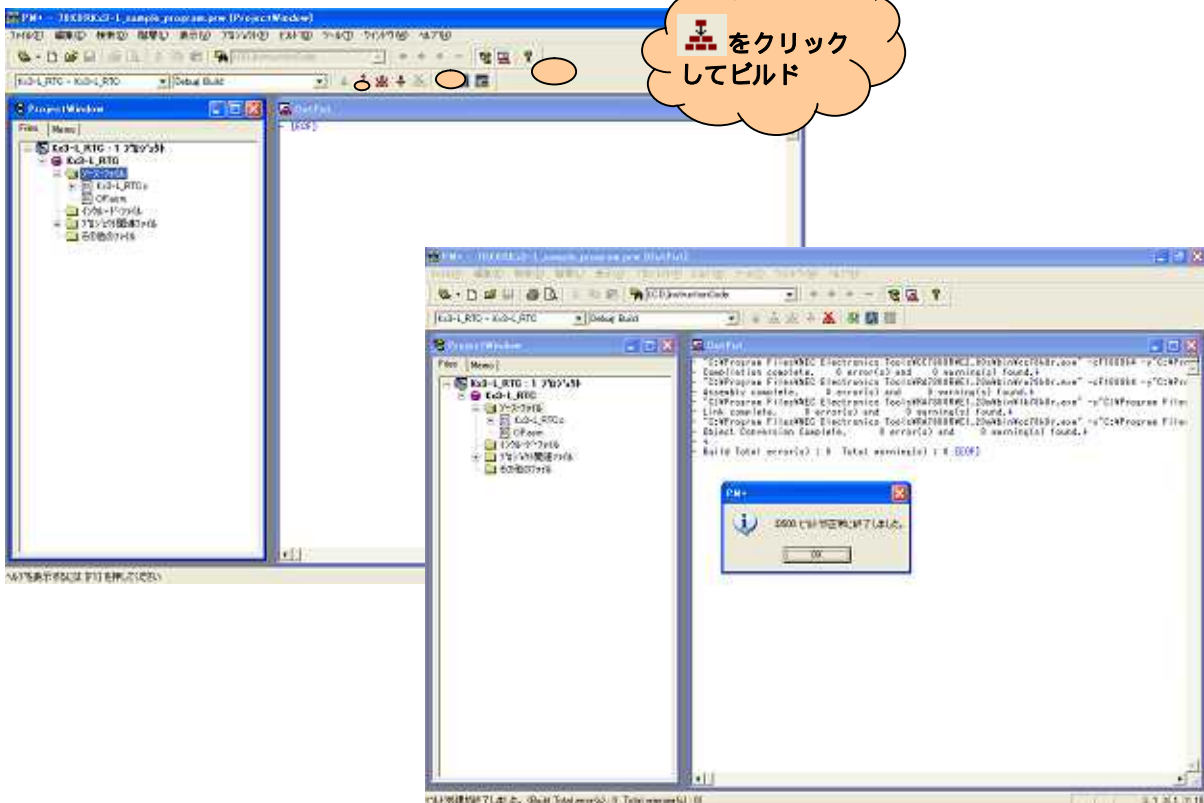
- (2) [プロジェクト] [プロジェクトの設定] を選択してください。[プロジェクトの設定] 画面が表示されたら、使用するデバイス名を選択（デフォルトでは、ROM/RAMサイズの最も大きいデバイスが選択）し、[OK] ボタンをクリックしてください



- (3) [ ツール ] [ オブジェクトコンバータオプションの設定 ] を選択してください。[ オブジェクトコンバータオプションの設定 ] 画面が表示されたら, [ 出力1 ] タグページが表示されているのを確認し, その中の出力ファイル名に任意のファイル名+拡張子 (.HEX) を入力し, [ OK ] をクリックします。



- (4) PM+画面の中央上付近の  をクリックしてください。自動でビルドが実行され, ソース・ファイルの「Kx3-L\_RTC.c」と「OP.asm」からHEXファイルが生成され, 「I3500:ビルドが正常に終了しました」というメッセージ画面が表示されます。  
[ OK ] をクリックしてビルドを終了します。



## 5.2 開発環境のダウンロード，インストール

78K0R/Kx3-Lマイクロコントローラの開発ツールのフリーツールは，次のサイトより入手可能です。

→<http://www.necel.com/micro/ja/freesoft/78k0r/index.html>

「RA78K0R」「CC78K0R」「78K0R/Kx3-L用デバイス・ファイル」の3ファイルをダウンロードし，インストールすることで，サンプル・プログラムの動作確認が可能となります。

ダウンロード，インストールは，上記サイトの画面および説明に従って，行ってください。

- 備考
1. PM+は，RA78K0Rに同封されています。
  2. ダウンロード後，登録したEメール・アドレスに，RA78K0R，CC78K0RのプロダクトIDが送付されます。このプロダクトIDは，各ツールのインストール時に必要となります。

## 第6章 関連資料

資料名		和文 / 英文
78K0R/Kx3-L ユーザーズ・マニュアル		<a href="#">PDF</a>
78K0Rシリーズ 命令編 ユーザーズ・マニュアル		<a href="#">PDF</a>
RA78K0R アセンブラ・パッケージ ユーザーズ・マニュアル	言語編	<a href="#">PDF</a>
	操作編	<a href="#">PDF</a>
CC78K0R Cコンパイラ ユーザーズ・マニュアル	言語編	<a href="#">PDF</a>
	操作編	<a href="#">PDF</a>
PM+ プロジェクト・マネージャ ユーザーズ・マニュアル		<a href="#">PDF</a>
SM+ システム・シミュレータ 操作編 ユーザーズ・マニュアル		<a href="#">PDF</a>

## 付録A プログラム・リスト

プログラム・リスト例として、78K0R/KE3-Lマイクロコントローラ用のソース・プログラムを次に示します。

Kx3-L\_RTC.asm (アセンブリ言語版)

```

*****
;
;
;   NEC Electronics      78K0R/KE3-Lシリーズ
;
;
*****
;   78K0R/KE3-Lシリーズ   サンプル・プログラム(リアルタイム・カウンタ)
*****
;   リアルタイム・カウンタ
*****
; 【履歴】
;   2009.01.-- 新規作成
*****
;
; 【概要】
; 本サンプル・プログラムでは、時計カウンタとして、年、月、曜日、日、時、分、秒を
; カウントするとともに、定周期割り込み、インターバル割り込み、アラーム割り込みに
; ついての実装例を示します。
;
; <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
;   ・高速内蔵発振回路に8MHz/20MHzを選択
;   ・LVIデフォルト・スタート機能動作
;   ・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
;   ・低電圧検出回路の機能を使用し、2.7V以上の電源電圧を確保
;   ・CPU/周辺ハードウェア・クロックを高速内蔵発振クロック動作の8MHzに設定
;   ・X1発振回路の停止
;
;
; <リアルタイム・カウンタの設定>
;   ・リアルタイム・カウンタへの入力クロックを供給
;   ・定周期割り込み(INTRTC)の周期を0.5秒に1度に設定
;   ・インターバル割り込み(INTRTC1)の周期を2^12/fXT(125ms)に設定

```

```

; ・現在日時を『2001/1/1(月) 00:00:00』に初期化
; ・リアルタイム・カウンタの動作開始
; ・定周期割り込み, インターバル割り込みの許可
;
;
; <メイン処理の主な内容>
; ・現在日時を『2008/4/10(日) 15:59:30』に設定
; ・アラーム設定を『毎日』, 『16:00』に設定
;
;
; <インターバル割り込み処理(INTRTCI)の主な内容>
; ・P11の出力を反転
; <定周期割り込み処理(INTRTC)の主な内容>
; ・P12の出力を反転
; <アラーム割り込み処理(INTRTC)の主な内容>
; ・16:00:00にP13をHigh出力
;
;
; <入出力ポートの設定>
; 入力ポート: -
; 出力ポート: P11-P13
; 未使用のポートで出力に設定できるものは全て出力ポートに設定しておく
;
;
; *****

```

```

;=====
;
; ベクタ・テーブルの設定
;
;=====

```

```

TVECT1          CSEG  AT   00000H
                DW    RESET_START          ;00000H    RESET入力,POC,LVI,WDT,TRAP
TVECT2          CSEG  AT   00004H
                DW    IINIT                ;00004H    INTWDTI
                DW    IINIT                ;00006H    INTLVI
                DW    IINIT                ;00008H    INTPO
                DW    IINIT                ;0000AH    INTP1
                DW    IINIT                ;0000CH    INTP2
                DW    IINIT                ;0000EH    INTP3
                DW    IINIT                ;00010H    INTP4
                DW    IINIT                ;00012H    INTP5

```



```

TVECT3          CSEG AT    00016H
                DW  IINIT          ;00016H   INTSR3
                DW  IINIT          ;00018H   INTSRE3
                DW  IINIT          ;0001AH   INTDMA0
                DW  IINIT          ;0001CH   INTDMA1
                DW  IINIT          ;0001EH   INTST0/INTCSI00
                DW  IINIT          ;00020H   INTSR0/INTCSI01
                DW  IINIT          ;00022H   INTSRE0
                DW  IINIT          ;00024H   INTST1/INTCSI10/INTIIC10
                DW  IINIT          ;00026H   INTSR1
                DW  IINIT          ;00028H   INTSRE1
                DW  IINIT          ;0002AH   INTIICA
                DW  IINIT          ;0002CH   INTTM00
                DW  IINIT          ;0002EH   INTTM01
                DW  IINIT          ;00030H   INTTM02
                DW  IINIT          ;00032H   INTTM03
                DW  IINIT          ;00034H   INTAD
                DW  IRTC          ;00036H   INTRTC
                DW  IRTC1        ;00038H   INTRTC1
                DW  IINIT          ;0003AH   INTKR

TVECT4          CSEG AT    00040H
                DW  IINIT          ;00040H   INTSRE2
                DW  IINIT          ;00042H   INTTM04
                DW  IINIT          ;00044H   INTTM05
                DW  IINIT          ;00046H   INTTM06
                DW  IINIT          ;00048H   INTTM07
                DW  IINIT          ;0004AH   INTP6
                DW  IINIT          ;0004CH   INTP7

```

=====

```

;
;
;   スタック領域の確保
;
;
;=====

```

```

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

```

```

XMAIN CSEG UNIT
*****
;

```

```

;
;   不要な割り込み要因による割り込み処理
;
;
;*****
;
IINIT:
;   不要な割り込みが発生した場合、ここに分岐します。
;   ここでは何も処理をしないで元の処理に戻ります

    RETI

;*****
;
;   リセット解除後の初期化処理
;
;*****
RESET_START:

;-----
;   割り込み禁止
;-----
    DI

;-----
;   レジスタ・バンク設定
;-----
    SEL    RBO

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

;-----
;   入出力ポートの設定
;-----
    CALL  !!SINIPOINT    ;出力に設定できるものは全て出力ポートに設定

;-----
;   低電圧検出
;-----
    CALL  !!SINILVI     ;2.7V以上の電源電圧を確保

;-----

```

; クロック周波数の設定

-----  
CALL !!SINICKL ;高速内蔵発振クロックを8MHzで動作

; リアルタイム・カウンタの動作設定

;リアルタイム・カウンタ初期設定

SET1 RTCEN ;リアルタイム・カウンタの入カクロック供給

MOV RTCC0, #00001001B ;リアルタイム・カウンタ・コントロール・レジスタ0

;||||+++----- CT2-CT0

;|||| [定周期割り込み (INTRTC) の選択]

;|||| 000:定周期割り込み機能を使用しない

;|||| 001:0.5秒に1度 (秒カウントアップに同期)

;|||| 010:1秒に1度 (秒カウントアップと同時)

;|||| 011:1分に1度 (毎分00秒)

;|||| 100:1時間に1度 (毎時00分00秒)

;|||| 101:1日に1度 (毎日00時00分00秒)

;|||| 11x:1月に1度 (毎月1日午前00時00分00秒)

;|||+----- AMPM

;||| [12時間制 / 24時間制の選択]

;||| 0:12時間制 (午前 / 午後を表示)

;||| 1:24時間制

;||+----- RCLOE0

;|| [RTCCL端子の出力制御]

;|| 0:RTCCL端子の出力(32.768kHz)禁止

;|| 1:RTCCL端子の出力(32.768kHz)許可

;||+----- RCLOE1

;|| [RTC1HZ端子の出力制御]

;|| 0:RTC1HZ端子の出力(1Hz)禁止

;|| 1:RTC1HZ端子の出力(1Hz)許可

;|+----- 必ず0に設定

;+----- RTCE

; [リアルタイム・カウンタの動作制御]

; 0:カウンタ動作停止

; 1:カウンタ動作開始

MOV RTCC1, #00000000B ;リアルタイム・カウンタ・コントロール・レジスタ1

;|||||+----- RWAIT

;||||| [リアルタイム・カウンタのウェイト制御]

;||||| 0:カウンタ動作設定

;||||| 1:SEC~YEARカウンタ停止設定。

```

;|||||          カウンタ値読み出し，書き込みモード。
;|||||+----- RWST
;|||||      [リアルタイム・カウンタのウェイト状態フラグ]
;|||||          0:カウンタ動作中
;|||||          1:カウンタ値の読み出し，書き込みモード中
;|||||+----- 必ず0に設定
;||||+----- RIFG
;||||      [定周期割り込みステータス・フラグ]
;||||          0:定周期割り込み発生なし
;||||          1:定周期割り込み発生あり
;|||+----- WAFG
;|||      [アラーム検出ステータス・フラグ]
;|||          0:アラーム不一致
;|||          1:アラームの一致検出
;||+----- 必ず0に設定
;|+----- WALIE
;|      [アラーム割り込み(INTRTC)機能の動作制御]
;|          0:アラームの一致による割り込みを発生しない
;|          1:アラームの一致による割り込みを発生する
;+----- WALE
;      [アラームの動作制御]
;          0:一致動作無効
;          1:一致動作有効

MOV  RTCC2, #1000111B ;リアルタイム・カウンタ・コントロール・レジスタ2
;+||||+++----- RINTE/ICT2-ICT0
; |||      [インターバル割り込み ( INTRTC1 ) の選択]
; |||          0xxx:インターバル割り込みを発生しない。
; |||          1000:2^6/fXT (1.953125 ms)
; |||          1001:2^7/fXT (3.90625 ms)
; |||          1010:2^8/fXT (7.8125 ms)
; |||          1011:2^9/fXT (15.625 ms)
; |||          1100:2^10/fXT (31.25 ms)
; |||          1101:2^11/fXT (62.5 ms)
; |||          111x:2^12/fXT (125 ms)
; ||++----- 必ず0に設定
; |+----- RCKDIV
; |      [RTCDIV端子の出力周波数の選択]
; |          0:RTCDIV端子から512Hzを出力 (1.95 ms)
; |          1:RTCDIV端子から16.384kHzを出力(0.061 ms)
; +----- RCLOE2
;      [RTCDIV端子の出力制御]
;          0:RTCDIV端子の出力禁止

```

```

;
; 1:RTCDIV端子の出力許可

;現在時刻初期化 (1/1/1(MON) 00:00:00)
MOV SEC, #00000000B ;秒カウント・レジスタ
;|+++++----- SEC40/SEC20/SEC10/SEC8/SEC4/SEC2/SEC1
;| 設定する値は, 10進の00-59をBCDコードで設定
;+----- 必ず0に設定

MOV MIN, #00000000B ;分カウント・レジスタ
;|+++++----- MIN40/MIN20/MIN10/MIN8/MIN4/MIN2/MIN1
;| 設定する値は, 10進の00-59をBCDコードで設定
;+----- 必ず0に設定

MOV HOUR, #00000000B ;時カウント・レジスタ
;||+++++----- HOUR20/HOUR10/HOUR8/HOUR4/HOUR2/HOUR1
;|| 設定する値は, 10進の00-23 (24時間制)
;|| または01-12, 21-32 (12時間制)をBCDコードで設定
;+----- 必ず0に設定

MOV WEEK, #00000001B ;曜日カウント・レジスタ
;||||+++----- WEEK4/WEEK2/WEEK1
;|||| 000:日曜日
;|||| 001:月曜日
;|||| 010:火曜日
;|||| 011:水曜日
;|||| 100:木曜日
;|||| 101:金曜日
;|||| 110:土曜日
;++++----- 必ず0に設定

MOV DAY, #00000001B ;日カウント・レジスタ
;||+++++----- DAY20/DAY10/DAY8/DAY4/DAY2/DAY1
;|| 設定する値は, 10進の01-31をBCDコードで設定
;+----- 必ず0に設定

MOV MONTH, #00000001B ;月カウント・レジスタ
;|||++++----- MONTH10/MONTH8/MONTH4/MONTH2/MONTH1
;||| 設定する値は, 10進の01-12をBCDコードで設定
;+++----- 必ず0に設定

MOV YEAR, #00000001B ;年カウント・レジスタ
;+++++----- YEAR80/YEAR40/YEAR20/YEAR10/YEAR8/YEAR4/YEAR2/YEAR1
; 設定する値は, 10進の00-99をBCDコードで設定

```

```

;リアルタイム・カウンタの割り込み許可
CLR1 RTCIF          ;INTRTC割り込み要求クリア
CLR1 RTCMK          ;INTRTC割り込み処理許可

CLR1 RTCIIF         ;INTRTCI割り込み要求クリア
CLR1 RTCIMK         ;INTRTCI割り込み処理許可

SET1 RTCE           ;カウンタ動作開始

```

```

-----
;
; 割り込み許可
; (割り込みを使用する場合はこのタイミングで許可します。)
;
-----

```

```

EI                  ;割り込み許可

BR  MAIN_LOOP      ;メイン・ループへ

```

```

*****
;
;
; 入出力ポートの設定
;
*****

```

SINIPORT:

```

-----
;
; デジタル入出力の設定
;
-----

```

```

MOV  ADPC, #0001000B          ;A/Dポート・コンフィギュレーション・レジスタ
      ;|||+++++----- ADPC4-ADPC0
      ;|||                  [アナログ入力(A) / デジタル入出力(D)の切り替え]
      ;|||                  +----+----- ANI11-ANI18/P153-P150
      ;|||                  |||+++++---- ANI7-ANI10/P27-P20
      ;|||                  0000:AAAAAAAAAA
      ;|||                  0001:AAAAAAAAAAD
      ;|||                  00010:AAAAAAAAAAD
      ;|||                  00011:AAAAAAAAADD
      ;|||                  00100:AAAAAAAADDDD
      ;|||                  00101:AAAAAADDDDD
      ;|||                  00110:AAAAAADDDDD
      ;|||                  00111:AAAAAADDDDD
      ;|||                  01000:AAAADDDDDDD
      ;|||                  01001:AAADDDDDDDDD
      ;|||                  01010:AADDDDDDDDD

```

```

;|||          01011:ADDDDDDDDDDD
;|||          10000:DDDDDDDDDDDD
;+++----- 必ず0に設定

```

```

;-----
;   ポート0の設定
;-----

```

```

MOV  P0, #00000000B      ;P00-P01の出力ラッチLow
MOV  PM0, #11111100B    ;P00-P01を出力ポートに設定
                          ;P00-P01:未使用

```

```

;-----
;   ポート1の設定
;-----

```

```

MOV  P1, #00000000B      ;P10-P17の出力ラッチLow
MOV  PM1, #00000000B    ;P10-P17を出力ポートに設定
                          ;P11:インターバル割り込み(INTRTCI)で使用する
                          ;P12:定周期割り込み(INTRTC)で使用する
                          ;P13:アラーム割り込み(INTRTC)で使用する
                          ;P10,P14-P17:未使用

```

```

;-----
;   ポート2の設定
;-----

```

```

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

```

```

;-----
;   ポート3の設定
;-----

```

```

MOV  P3, #00000000B      ;P30-P33の出力ラッチLow
MOV  PM3, #11110000B    ;P30-P33を出力ポートに設定
                          ;P30-P33:未使用

```

```

;-----
;   ポート4の設定
;-----

```

```

MOV  P4, #00000000B      ;P40-P43の出力ラッチLow
MOV  PM4, #11110000B    ;P40-P43を出力ポートに設定
                          ;P40-P43:未使用

```

; ポート5の設定

-----

MOV P5, #00000000B ;P50-P53の出力ラッチLow  
 MOV PM5, #111110000B ;P50-P53を出力ポートに設定  
 ;P50-P53:未使用

-----

; ポート6の設定

-----

MOV P6, #00000000B ;P60-P61の出力ラッチLow  
 MOV PM6, #11111100B ;P60-P61を出力ポートに設定  
 ;P60-P61:未使用

-----

; ポート7の設定

-----

MOV P7, #00000000B ;P70-P77の出力ラッチLow  
 MOV PM7, #00000000B ;P70-P77を出力ポートに設定  
 ;P70-P77:未使用

-----

; ポート8の設定

-----

MOV P8, #00000000B ;P80-P83の出力ラッチLow  
 MOV PM8, #111110000B ;P80-P83を出力ポートに設定  
 ;P80-P83:未使用

-----

; ポート12の設定

-----

MOV P12, #00000000B ;P120の出力ラッチLow  
 MOV PM12, #11111110B ;P120を出力ポートに設定  
 ;P121,P122:未使用  
 ;P123,P124:XT1/XT2入力端子  
 ; P121-P124は入力ポート

-----

; ポート14の設定

-----

MOV P14, #00000000B ;P140-P141の出力ラッチLow  
 MOV PM14, #11111100B ;P140-P141を出力ポートに設定  
 ;P140-P141:未使用



```

;-----
;   ポート15の設定
;-----

```

```

MOV   P15, #0000000B           ;P150-P153の出力ラッチLow
MOV   PM15, #11110000B        ;P150-P153を出力ポートに設定
                                       ;P150-P153:未使用

RET

```

```

;*****
;
;

```

```

;   低電圧検出
;

```

```

;-----
;   低電圧検出回路の機能を使用し, 2.7V以上の電源電圧を確保します。
;*****

```

SINILVI:

```

;低電圧検出回路の設定

```

```

SET1  LVIMK                   ;INTLVI 割り込み禁止
CLR1  LVISEL                  ;検出電圧をVDDに設定
MOV   LVIS, #00001001B ;低電圧検出レベル選択レジスタ
                                       ;||||+---- LVIS3-LVIS0
                                       ;||||      [検出レベル]
                                       ;||||      0000:VLVI0 (4.22±0.1V)
                                       ;||||      0001:VLVI1 (4.07±0.1V)
                                       ;||||      0010:VLVI2 (3.92±0.1V)
                                       ;||||      0011:VLVI3 (3.76±0.1V)
                                       ;||||      0100:VLVI4 (3.61±0.1V)
                                       ;||||      0101:VLVI5 (3.45±0.1V)
                                       ;||||      0110:VLVI6 (3.30±0.1V)
                                       ;||||      0111:VLVI7 (3.15±0.1V)
                                       ;||||      1000:VLVI8 (2.99±0.1V)
                                       ;||||      1001:VLVI9 (2.84±0.1V)
                                       ;||||      1010:VLVI10(2.68±0.1V)
                                       ;||||      1011:VLVI11(2.53±0.1V)
                                       ;||||      1100:VLVI12(2.38±0.1V)
                                       ;||||      1101:VLVI13(2.22±0.1V)
                                       ;||||      1110:VLVI14(2.07±0.1V)
                                       ;||||      1111:VLVI15(1.91±0.1V)
                                       ;++++----- 必ず0に設定
CLR1  LVIMD                   ;低電圧検出時の動作モードを割り込み信号発生に設定
SET1  LVION                   ;低電圧検出動作許可

```

```

;低電圧検出回路の動作安定待ち(約10us)

```

```

MOV B, #10 ;カウント回数設定
HRES100:
NOP ; (1clk)
DEC B ; (1clk)
BNZ $HRES100 ;ウエイト完了? No, (2clk/4clk)

```

;VLVI VDDになるまでのウエイト

```

HRES300:
NOP
BT LVIF, $HRES300 ;VDD < VLVI? Yes,
CLR1 LVION ;低電圧検出動作停止

RET

```

```

;*****
;
;
; クロック周波数の設定
;
;-----
;
; 高速内蔵発振クロックで動作が行えるように設定します。
;*****

```

```

SINICLK:
MOV CMC, #00010000B ;クロック動作モード
;|||||+----- AMPH
;||||| [高速システム・クロック発振周波数の制御]
;||||| 0: 2MHz fMX < 10MHz
;||||| 1: 10MHz < fMX 20MHz
;||||+----- AMPHS1-AMPHSO
;|||| [XT1発振回路の発振モード選択]
;|||| 00: 低消費発振 (デフォルト)
;|||| 01: 通常発振
;|||| 10: 超低消費発振
;|||| 11: 超低消費発振
;|||+----- 必ず0に設定
;|||+----- OSCSELS
;||| [サブシステム・クロック端子の動作モード]
;||| 0: 入力ポート・モード
;||| 1: XT1発振モード
;||+----- 必ず0に設定
;++----- EXCLK/OSCSEL
; [高速システム・クロック端子の動作モード]
;
; 00: 入力ポート・モード
;
; 01: XT1発振モード

```

```

;
; 10:入力ポート・モード
;
; 11:外部クロック入力モード

MOV  CSC, #10000000B ;クロック動作ステータス制御
;| | | | | | | | | | +----- HI0STOP
;| | | | | | | | | | [高速内蔵発振クロックの動作制御]
;| | | | | | | | | | 0:高速内蔵発振回路動作
;| | | | | | | | | | 1:高速内蔵発振回路停止
;| | +----- 必ず0に設定
;| +----- XTSTOP
;| [サブシステム・クロックの動作制御]
;| 0:XT1発振回路動作
;| 1:XT1発振回路停止
;+----- MSTOP
; [高速システム・クロックの動作制御]
;
; 0:X1発振回路動作
;
; 1:X1発振回路停止

MOV  OSMC, #10000000B ;動作スピード・モード
;| | | | | | | | | | +----- FSEL/FLPC
;| | | | | | | | | | [fCLKの周波数選択]
;| | | | | | | | | | 00:10MHz以下の周波数で動作(デフォルト)
;| | | | | | | | | | 01:10MHzを越える周波数で動作
;| | | | | | | | | | 10:1MHzの周波数で動作
;| | | | | | | | | | 11:設定禁止
;| | +----- 必ず0に設定
;| +----- RTCLPC
; [サブシステム・クロックHALTモード時の設定]
;
; 0:周辺機能へのサブシステム・クロック供給許可
;
; 1:リアルタイム・カウンタ以外の周辺機能への
;
; サブシステム・クロック供給停止

MOV  CKC, #00001001B ;クロック選択
;| + | + | +----- CSS/MCM0/MDIV2-MDIV0
;| | | [CPU/周辺ハードウェア・クロック(fCLK)の選択]
;| | | 00000:fIH
;| | | 00001:fIH/2(デフォルト)
;| | | 00010:fIH/2^2
;| | | 00011:fIH/2^3
;| | | 00100:fIH/2^4
;| | | 00101:fIH/2^5
;| | | 01000:fMX
;| | | 01001:fMX/2

```

```

;| | |          01010: fMX/2^2
;| | |          01011: fMX/2^3
;| | |          01100: fMX/2^4
;| | |          01101: fMX/2^5
;| | |          1xxxx: fSUB/2
;| | +----- 必ず1に設定
;| +----- MCS <Read Only>
;|             [メイン・システム・クロック (fMAIN)のステータス]
;|             0: 高速内蔵発振クロック (fIH)
;|             1: 高速システム・クロック (fMX)
;+----- CLS <Read Only>
;|             [CPU/周辺ハードウェア・クロック (fCLK)のステータス]
;|             0: メイン・システム・クロック (fMAIN)
;|             1: サブシステム・クロック (fSUB)

MOV   DSCCTL,   #0000000B      ;20MHz高速内蔵発振制御
;| | | | | +----- DSCON
;| | | | |      [20MHz高速内蔵発振クロック (fIH20)の動作許可/禁止]
;| | | | |      0: 動作禁止
;| | | | |      1: 動作許可
;| | | | +----- 必ず0に設定
;| | | | +----- SELDSC
;| | | |      [CPU/周辺ハードウェア・クロック (fCLK)への
;| | | |                                     20MHz高速内蔵発振選択]
;| | | |      0: 20MHz高速内蔵発振を選択しない
;| | | |      1: 20MHz高速内蔵発振を選択
;| | | +----- DSCS <Read Only>
;| | |      [20MHz高速内蔵発振供給状態フラグ]
;| | |      0: 供給していない
;| | |      1: 供給している
;+ + + + +----- 必ず0に設定

RET

;
;
;   メイン処理
;
;
;*****
;
MAIN_LOOP:
MOV   SUBCUD,   #0000000B      ;時計誤差補正レジスタ (時計誤差補正無しに設定)
;| + + + + +----- F6-F0
;|             x00000x: 時計誤差補正無し

```

```

;|          0xxxxxx: ((F5,F4,F3,F2,F1,F0)-1)*2だけ増加
;|          1xxxxxx: ((/F5,/F4,/F3,/F2,/F1,/F0)+1)*2だけ減少
;|          /F5~/F0はF5~F0のビット反転した値
;+----- DEV
;          0: 秒桁が00,20,40秒時(20秒ごと)に時計誤差補正
;          1: 秒桁が00秒時のみ(60秒ごと)に時計誤差補正

;-- カウンタ停止 --
SET1  RWAIT          ;カウンタ停止設定

LMAIN100:
    BF  RWST, $LMAIN100      ;カウンタ停止待ち
;-----
;  現在時刻設定
;  2008/4/10(SUN) 15:59:30
;-----
    MOV  SEC, #30H          ;秒カウント・レジスタを30秒に設定
    MOV  MIN, #59H          ;分カウント・レジスタを59分に設定
    MOV  HOUR, #15H         ;時カウント・レジスタを15時に設定
    MOV  WEEK, #00H         ;曜日カウント・レジスタを日曜日に設定
    MOV  DAY, #10H          ;日カウント・レジスタを10日に設定
    MOV  MONTH, #04H        ;月カウント・レジスタを4月に設定
    MOV  YEAR, #08H         ;年カウント・レジスタを08年に設定

;-- カウンタ動作 --
CLR1  RWAIT          ;カウンタ動作設定

LMAIN200:
    BT  RWST, $LMAIN200      ;カウンタ動作待ち
;-----
;  アラーム設定
;  毎日 16:00
;-----
    CLR1  WALE          ;一致動作無効
                    ;アラームの各レジスタ(WALIE,ALARMMM,ALARMWH,
                    ;ALARMWW)設定のため

    MOV  ALARMWH,#00010110B   ;アラーム時レジスタを16時に設定
        ;|+----- WH20/WH10/WH8/WH4/WH2/WH1
        ;|          設定する値は、10進の00-23(24時間制)または
        ;|          01-12,21-32(12時間制)をBCDコードで設定
        ;+----- 必ず0に設定

    MOV  ALARMMM,#00000000B   ;アラーム分レジスタを00分に設定

```

```

;|+++++----- WM40/WM20/WM10/WM8/WM4/WM2/WM1
;|          設定する値は, 10進の00-99をBCDコードで設定
;+----- 必ず0に設定

```

```

MOV  ALARMWW,#0111111B      ;アラーム週レジスタを毎日に設定
;|+++++----- WW0
;|+++++          0:日曜日アラームOFF設定
;|+++++          1:日曜日アラームON設定
;|+++++----- WW1
;|+++++          0:月曜日アラームOFF設定
;|+++++          1:月曜日アラームON設定
;|+++++----- WW2
;|+++++          0:火曜日アラームOFF設定
;|+++++          1:火曜日アラームON設定
;|+++----- WW3
;|+++           0:水曜日アラームOFF設定
;|+++           1:水曜日アラームON設定
;|++----- WW4
;|++            0:木曜日アラームOFF設定
;|++            1:木曜日アラームON設定
;|+----- WW5
;|+             0:金曜日アラームOFF設定
;|+             1:金曜日アラームON設定
;|+----- WW6
;|              0:土曜日アラームOFF設定
;|              1:土曜日アラームON設定
;+----- 必ず0に設定

```

```

SET1  WALIE      ;アラーム割込有効設定
SET1  WALE       ;一致動作有効

```

```

;-----
; 無限ループ
;-----

```

```

LMAIN500:
    HALT
    BR   LMAIN500

```

```

*****
;
;
;   アラーム処理
;
*****
;

```

SALAM:

;アラーム発生時の処理をここに記述

RET

```

;*****
;
;
; 定周期割込処理 , アラーム割込処理 (INTRTC)
;
;*****

```

IRTC:

```

SEL  RB1                ;レジスタ・バンク設定

BF   RIFG, $HRTC500    ;RTC定周期割込発生? No, (アラーム割込判定)
XOR  P1, #00000100B   ; Yes, P12の出力を反転する

```

;定周期割り込みステータス・フラグのクリア

```

MOV  A, RTCC1
AND  A, #11010011B     ;定周期割り込みクリア
; | | +----- 必ず0に設定
; | +----- RIFGのクリア
; +----- 必ず0に設定
MOV  RTCC1, A

```

HRTC500:

```

BF   WAFG, $HRTC900   ;アラーム割込発生? No, (何もしない)
SET1 P1.3             ; Yes, アラーム処理開始
CALL !SALAM           ;アラーム発生時の処理

```

;アラーム検出ステータス・フラグのクリア

```

MOV  A, RTCC1
AND  A, #11001011B    ;アラームの一致検出クリア
; || +----- 必ず0に設定
; |+----- WAFGのクリア
; +----- 必ず0に設定
MOV  RTCC1, A

```

HRTC900:

RETI

```

;*****
;
;

```

```
;   インターバル割込処理(INTRTCI)
;
;
;*****
;
IRTCI:
    XOR   P1,   #00000010B   ;P11の出力を反転する

    RETI

end
```



Kx3-L\_RTC.c (C言語版)

/\*

NEC Electronics 78K0R/Kx3-Lシリーズ

\*/

78K0R/Kx3-Lシリーズ サンプル・プログラム

\*/

リアルタイム・カウンタ

\*/

【履歴】

2008.11.16 新規作成

2009.01.26 処理変更

\*/

【概要】

時計カウンタとして、年、月、曜日、日、時、分、秒をカウントします。  
 定周期割り込み、インターバルタイマ割り込み、アラーム割り込みについての実装例を示します。  
 定周期割り込みは、0.5秒間隔で発生します。  
 インターバルタイマ割り込みは、125ミリ秒間隔で発生します。  
 アラーム割り込みは、現在日時8/4/10 15:59:30の設定に対して16:00丁度に割り込みが発生します。

<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
- ; ・高速内蔵発振回路に8MHz/20MHzを選択
- ; ・LVIデフォルト・スタート機能動作
- ; ・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
- ; ・低電圧検出回路の機能を使用し、2.7V以上の電源電圧を確保
- ; ・CPU/周辺ハードウェア・クロックを高速内蔵発振クロック動作の8MHzに設定
- ; ・X1発振回路の停止
- ;
- ;
- ;

<リアルタイム・カウンタの設定>

- ; ・リアルタイム・カウンタへの入力クロックを供給
- ; ・定周期割り込み(INTRTC)の周期を0.5秒に1度に設定
- ; ・インターバル割り込み(INTRTC1)の周期を2<sup>12</sup>/fXT(125ms)に設定
- ; ・現在日時を『2001/1/1(月) 00:00:00』に初期化
- ; ・リアルタイム・カウンタの動作開始

```

; ・定周期割り込み，インターバル割り込みの許可
;
;
; <メイン処理の主な内容>
; ・現在日時を『2008/4/10(日) 15:59:30』に設定
; ・アラーム設定を『毎日』，『16:00』に設定
;
;
; <インターバル割り込み処理(INTRTCI)の主な内容>
; ・P11の出力を反転
; <定周期割り込み処理(INTRTC)の主な内容>
; ・P12の出力を反転
; <アラーム割り込み処理(INTRTC)の主な内容>
; ・16:00:00にP13をHigh出力
;
;
; <入出力ポートの設定>
; 入力ポート：-
; 出力ポート：P11-P13
;   インターバルタイマ割り込み発生の際にトグル出力：P1.1
;   定周期割り込み発生の際にトグル出力：P1.2
;   アラーム割り込み発生でHIGH出力：P1.3
; 未使用のポートで出力に設定できるものは全て出力ポートに設定しておく

*****/

/*=====
   前処理指令（#pragma指令）
=====*/
#pragma SFR /* 特殊機能レジスタ(SFR)名を記述可能にする */
#pragma NOP /* NOP()を記述可能にする */
#pragma HALT /* HALT()を記述可能にする */
#pragma DI /* DI()を記述可能にする */
#pragma EI /* EI()を記述可能にする */

/*=====
   割り込みハンドラ定義
=====*/
#pragma interrupt INTRTC INTRTC_hdr rb1 /* RTCアラーム/定周期割込 */
#pragma interrupt INTRTCI INTRTCI_hdr /* RTCインターバル割込 */

/*=====

```

関数プロトタイプ宣言

```

=====*/
void hdwinit( void );           /* 機能初期化 */
static void f_ini_cpu(void);    /* CPU初期設定 */
static void f_ini_lvi(void);    /* 電源電圧立ち上がり待ち */
static void f_ini_port(void);   /* ポート初期設定 */
static void f_ini_reg(void);    /* 周辺レジスタ初期設定 */
static void f_ini_itr(void);    /* 割り込み初期設定 */

void salarm(void);             /* アラーム発生時の処理関数 */
void main(void);              /* アプリケーションメイン */

```

```

/*=====
グローバル変数の定義
=====*/

```

```

/*****
* Title : リセット解除後の初期化処理
*****
* Module : void hdwinit(void)
* Arg   :
* Ret   :
* -----
* Note  :
*****/

```

```

void hdwinit(void)
{
    f_ini_port();           /* ポート初期設定 */
    f_ini_lvi();           /* 2.7V以上の電源電圧を確保 */
    f_ini_cpu();          /* CPU初期設定 */
    f_ini_reg();          /* 周辺レジスタ初期設定 */
    f_ini_itr();          /* 割り込み初期設定 */
}

```

```

/*****
* Title : 電源電圧検出
*****
* Module : static void f_ini_lvi(void)
* Arg   :
* Ret   :
* -----
* Note : LVI機能を使い電源電圧が2.7V以上になるのを待ちます。
*****/

```

```

static void f_ini_lvi(void)
{
    unsigned char ucCnt10us;          /* LVI起動待ち時間計測用 */
/*-----
;    低電圧検出回路の機能を使用し, 2.7V以上の電源電圧を確保します。
;-----*/

    LVIMK = 1;                        /* INTLVI割り込み禁止 */
    LVIM = 0b00000000;                /* 低電圧検出レジスの設定 */
/*
    |||||+--LVIFフラグ
    |||||
    |||||+---LVIMD: 低電圧検出の動作モード選択
    |||||          0 : 割り込みモード
    |||||          1 : リセット・モード
    |||||
    |||||+-----LVISEL: 電圧検出の選択
    |||||          0 : 電源電圧(VDD)のレベルを検出
    |||||          1 : 外部入力端子からの入力電圧のレベルを検出
    |||||
    |++++----- 必ず0に設定
    |
    +-----LVION: 低電圧検出動作許可
    0 : 動作禁止
    1 : 動作許可

*/

    LVIS = 0b00001001;                /* 検出電圧の設定 */
/*
    ||||+----LVIS3-0: [検出レベル]
    ||||          0000 : 4.22 ± 0.1V
    ||||          0001 : 4.07 ± 0.1V
    ||||          0010 : 3.92 ± 0.1V
    ||||          0011 : 3.76 ± 0.1V
    ||||          0100 : 3.61 ± 0.1V
    ||||          0101 : 3.45 ± 0.1V
    ||||          0110 : 3.30 ± 0.1V
    ||||          0111 : 3.15 ± 0.1V
    ||||          1000 : 2.99 ± 0.1V
    ||||          1001 : 2.84 ± 0.1V
    ||||          1010 : 2.68 ± 0.1V
    ||||          1011 : 2.53 ± 0.1V
    ||||          1100 : 2.38 ± 0.1V
    ||||          1101 : 2.22 ± 0.1V
    ||||          1110 : 2.07 ± 0.1V

```

```

        ||||                1111 : 1.91 ± 0.1V
        ||||
        ++++----- 必ず0に設定
*/

LVION =    1;                /* 低電圧検出動作許可 */

/*
    低電圧検出回路の動作安定待ち(約10us)
*/
for( ucCnt10us = 0; ucCnt10us < 3; ucCnt10us++ ){
    NOP();
    NOP();
}

/*
    VLVI  VDDになるまでのウェイト
*/
while(LVIF){
    NOP();
}
LVION =    0;                /* 低電圧検出動作停止 */
}

/*****
* Title   : CPU初期設定
*****
* Module  : static void f_ini_cpu(void)
* Arg     :
* Ret     :
*-----
* Note    :
*****/

static void f_ini_cpu(void)
{
    CMC =    0b00010000;        /* クロック動作モード制御レジスタ(CMC) */
/*
    ||||||+-- AMPH: 高速システム・クロック発振周波数の制御
    ||||||                0 : 2MHz<=fMX<=10MHz
    ||||||                1 : 10MHz<=fMX<=20MHz
    ||||||
    |||||+---- AMPHS1, AMPS0: XT1発振回路の発振モード選択
    |||||                00 : 低消費発振(デフォルト)
    |||||                01 : 通常発振

```

```

|||||          1x : 超低消費発振
|||||
||+|+----- 必ず0に設定
|| |
|| +-----OSCSELS: サブシステム・クロック端子の動作モード
||          0 : 入力ポート・モード
||          1 : XT1発振モード
++-----EXCLK, OSCSEL: 高速システム・クロック端子の動作モード
          00 : 入力ポート・モード
          01 : X1発振モード
          10 : 入力ポート・モード
          11 : 外部クロック入力モード

*/

CSC =          0b10000000;          /* クロック動作ステータス制御レジスタ(CSC) */
/*
|||||+---HIOSTOP: 高速内蔵発振クロックの動作制御
|||||          0 : 高速内蔵発振回路動作
|||||          1 : 高速内蔵発振回路停止
|||||
||+++++--- 必ず0に設定
||
|+-----XTSTOP: サブシステム・クロックの動作制御
|          0 : XT1発振回路動作
|          1 : XT1発振回路停止
|
+-----MSTOP: 高速システム・クロックの動作制御
          0 : X1発振回路動作
          1 : X1発振回路停止

*/

CKC =          0b00001000;          /* システム・クロック制御レジスタ(CKC)          */
/*
|+|+|+++--CSS, MCM0, MDIV2, MDIV1, MDIV0:
| | |          CPU/周辺ハードウェア・クロックの選択
| | |          00000 : f1H
| | |          00001 : f1H/2
| | |          00010 : f1H/4
| | |          00011 : f1H/8
| | |          00100 : f1H/16
| | |          00101 : f1H/32
| | |          01000 : fMX
| | |          01001 : fMX/2
| | |          01010 : fMX/4
| | |          01011 : fMX/8

```

```

| | |          01100 : fMX/16
| | |          01101 : fMX/32
| | |          1xxxx : fSUB/2
| | |
| | +----- 必ず1に設定
| |
| +-----MCS: メイン・システム・クロック (fMAIN)のステータス
|          0 : 高速内蔵発振クロック (fIH)
|          1 : 高速システム・クロック (fMX)
|
+-----CLS: CPU/周辺ハードウェア・クロック (fCLK)のステータス
          0 : メイン・システム・クロック (fMAIN)
          1 : サブシステム・クロック (fSUB)
*/

DSCCTL = 0b00000000; /* 6) 20 MHz高速内蔵発振制御レジスタ (DSCCTL) */
/*      |||||+--DSCON: 20 MHz高速内蔵発振クロックの動作許可/禁止
|||      0 : 動作禁止
|||      1 : 動作許可
|||
++++|+--- 必ず0に設定
      ||
      |+----SELDSC: CPU/周辺ハードウェア・クロック (fCLK) への20 MHz高速内蔵発振選択
      |          0 : 20 MHz高速内蔵発振を選択しない
      |          1 : 20 MHz高速内蔵発振を選択
      |
      +-----20 MHz高速内蔵発振供給状態フラグ
*/

/*-----
;   周辺機能へのクロック供給 (イネーブル・レジスタ)
;-----*/

/*

この段階では周辺機器へのクロック供給は禁止しておく

*/

PERO = 0b00000000; /* 周辺イネーブル・レジスタ0(PERO) */
/*      +|+|+|+--- 必ず0に設定
      | | |
      | | +----SAU0EN: シリアル・アレイ・ユニットの入力クロックの制御

```

```

| ||          0 : 入力クロック供給停止
| ||          1 : 入力クロック供給
| ||
| |+-----||CAEN: シリアル・インターフェース||CAの入力クロックの制御
| |          0 : 入力クロック供給停止
| |          1 : 入力クロック供給
| |
| +-----ADCEN: A/Dコンバータの入力クロックの制御
|           0 : 入力クロック供給停止
|           1 : 入力クロック供給
|
+-----RTCEN: リアルタイム・カウンタ(RTC)の入力クロックの制御
          0 : 入力クロック供給停止
          1 : 入力クロック供給
*/

```

```

PER1 = 0b00000000; /* 周辺イネーブル・レジスタ1(PER1) */
/*
++++|+++-- 必ず0に設定
+-----OACMPEN: オペアンプの入力クロックの制御
          0 : 入力クロック供給停止
          1 : 入力クロック供給
*/

```

```

PER2 = 0b00000000; /* 周辺イネーブル・レジスタ2(PER2) */
/*
+++++++|-- 必ず0に設定
|
+---TAUOEN: タイマ・アレイ・ユニットTAUSの入力クロックの制御
          0 : 入力クロック供給停止
          1 : 入力クロック供給
*/

```

```

/*
/*-----
;   動作スピード・モード制御レジスタ(OSMC)
;-----*/

```

設定はデフォルトのままでもいいので設定は省略

```

/*
OSMC = 0b00000000; /* 動作スピード・モード制御レジスタ(OSMC) */
/*
|+++++|-- 必ず0に設定
|   ||

```



```

|    ++--FLPC,FSEL: fCLKの周波数選択
|
|           00 : 10MHz以下の周波数で動作(デフォルト)
|           01 : 10MHzを越える周波数で動作
|           10 : 1MHz以下の周波数で動作
|           11 : 設定禁止
|
+-----RTCLPC: サブシステム・クロックHLATモード時の設定
           0 : 周辺機能へのサブシステム・クロック供給許可
           1 : RTC以外の周辺機能へのサブシステム・クロック供給停止
*/

}

/*****
* Title   : ポート初期設定
*****

* Module : static void f_ini_port(void)
* Arg    :
* Ret    :
*-----

* Note   :
*****/

static void f_ini_port(void)
{
/*****
;   ポート0の設定(未使用)
;
;
;*****/

    P0 = 0b00000000;          /* P01,P00を0に設定 */
    PM0 = 0b11111100;        /* P01,P00を出力に設定 */

/*****
;   ポート1の設定
;
;   P11-P13をLEDドライブで使用する。他の未使用端子も出力に設定する。
;
;*****/

    P1 = 0b00000110;          /* LED出力以外は全て0に設定する */
    PM1 = 0b00000000;          /* P17~P10を出力に設定する */
/*****
|+---P11 : LED1 (PWM出力) 用出力ポート
|+---P12 : LED2 (更新) 用出力ポート

```

```

+-----P13 : アラーム発生出力

*/
/*****
;   ポート 2 の設定 (未使用)
;
;*****/
/* ADPCは設定しないが参考として示す
   ADPC =      0b00010000;*/           /* A/Dポート・コンフィギュレーション・
                                       レジスタ (ADPC) : 全てデジタルに */

   P2 = 0b00000000;                   /* 全ての出力ラッチを0に設定 */
   PM2 = 0b00000000;                  /* 全てのビットを出力に設定 */

/*****
;   ポート 3 の設定 (未使用)
;
;*****/
/* PIM3とPOM3は設定しないが参考として示す
   PIM3 =      0b00000000;*/          /* P32 ~ P33は通常入力 */
/* POM3 =      0b00000000;*/          /* P32 ~ P33は通常出力モード */
   P3 = 0b00000000;                   /* P33 ~ P30の出力ラッチを0に設定 */
   PM3 = 0b11110000;                  /* P33 ~ P30は出力に設定 */

/*****
;   ポート 4 の設定 (未使用)
;
;*****/

   P4 = 0b00000000;                   /* P43 ~ P40の出力ラッチを0に設定 */
   PM4 = 0b11110000;                  /* 全てのビットを出力に設定 */

/*****
;   ポート 5 の設定 (未使用)
;
;*****/

   P5 = 0b00000000;                   /* P53 ~ P50の出力ラッチを0に設定 */
   PM5 = 0b11110000;                  /* 全てのビットを出力に設定 */

/*****
;   ポート 6 の設定 (未使用)
;
;*****/

```

```

P6 = 0b00000000;          /* P61 ~ P60の出力ラッチを0に設定 */
PM6 = 0b11111100;        /* 全てのビットを出力に設定 */

/*****
;   ポート 7 の設定 (未使用)
;
;
; *****/
/* PIM7とPOM7は設定しないが参考として示す
   PIM7 = 0b00000000; /* P75 ~ 4、P72 ~ 1は通常入力 */
/* POM7 = 0b00000000; /* P75、P73 ~ 2、P70は通常出力モード */
P7 = 0b00000000;      /* P77 ~ P70の出力ラッチを0に設定 */
PM7 = 0b00000000;    /* 全てのビットを出力に設定 */

/*****
;   ポート 8 の設定 (未使用)
;
;
; *****/
/* PIM8は設定しないが参考として示す
   PIM8 = 0b00001111; /* P83 ~ P80はデジタル入力許可 */
P8 = 0b00000000;    /* P83 ~ P80の出力ラッチを0に設定 */
PM8 = 0b11110000;  /* 全てのビットを出力に設定 */

/*****
;   ポート 1 2 の設定 (未使用)
;
;
; *****/

P12 = 0b00000000;     /* P124 ~ P120の出力ラッチを0に設定 */
PM12 = 0b11111100;   /* P120を出力に設定する */

/*****
;   ポート 1 4 の設定 (未使用)
;
;
; *****/

P14 = 0b00000000;     /* P61 ~ P60の出力ラッチを0に設定 */
PM14 = 0b11111100;   /* 全てのビットを出力に設定 */

/*****
;   ポート 1 5 の設定 (未使用)
;
;
; *****/

```

```

P15 = 0b00000000;          /* P153~P150の出力ラッチを0に設定 */
PM15 = 0b11110000;        /* 全てのビットを出力に設定 */

}

/*****
* Title   : 周辺レジスタ初期設定
*****

* Module  : static void f_ini_reg(void)
* Arg     :
* Ret     :
* -----
* Note    :
*****/

static void f_ini_reg(void)
{
/*****
-   RTCへの入力クロック供給
-   10000000 (PER0:周辺イネーブル・レジスタ0)
-   ||| |||
-   ||| |||+--- 0
-   ||| |||+---- 0
-   ||| |||+----- SAU0EN
-   ||| |||+----- 0
-   ||| |||+----- IICAEN
-   ||| |||+----- ADCEN
-   ||| |||+----- 0
-   ||| |||+----- RTCEN (RTC > 0:供給停止 / 1:入力クロック供給)
;*****/

    RTCEN = 1;              /* 入力クロック供給 */

/*****
-   リアルタイム・カウンタ・コントロール・レジスタ0
;*****/

    RTCC0 = 0b00001001;     /* (RTCC0:リアルタイム・カウンタ・コントロール・レジスタ0)
-   ||| |||
-   ||| |||+++--- CT2-CT0[定周期割り込み (INTRTC) の選択]
-   ||| |||          000 : 定周期割り込み機能を使用しない
-   ||| |||          001 : 0.5秒に1度 (秒カウントアップに同期)
-   ||| |||          010 : 1秒に1度 (秒カウントアップと同時)

```

```

- |||||          011 : 1分に1度 ( 毎分00秒 )
- |||||          100 : 1時間に1度 ( 毎時00分00秒 )
- |||||          101 : 1日に1度 ( 毎日00時00分00秒 )
- |||||          11x : 1月に1度 ( 毎月1日午前00時00分00秒 )
- |||||
- |||||+----- AMPM[12時間制 / 24時間制の選択]
- ||||          0 : 12時間制
- ||||          1 : 24時間制
- ||||
- |||+----- RCLOE0[RTCCL端子の出力制御]
- |||          0 : 32.768kHz出力禁止
- |||          1 : 32.768kHz出力許可
- |||
- ||+----- RCLOE1 [RTCCL端子の出力制御]
- ||          0 : 1kHz出力禁止
- ||          1 : 1kHzの出力許可
- ||
- |+----- 必ず0に設定
- |
- +----- RTCE[リアルタイム・カウンタの動作制御]
          0 : カウンタ動作停止
          1 : カウンタ動作開始

```

\*/

```

/* RTCE   : カウンタ動作停止 */
/* RCLOE1: RTC1HZ端子の出力(1Hz)禁止 */
/* RCLOE0: RTCCL端子の出力(32kHz)禁止 */
/* AMPM   : 24時間制 */
/* CTn    : 周期1000ms */

```

/\*-----\*/

- リアルタイム・カウンタ・コントロール・レジスタ1

/\*-----\*/

```

RTCC1 = 0b00000000; /* (RTCC1:リアルタイム・カウンタ・コントロール・レジスタ1)
- |||||+----- RWAIT[リアルタイム・カウンタのウェイト制御]
- |||||          0 : カウンタ動作設定
- |||||          1 : SEC~YEARカウンタ停止設定。カウンタ値読み出し
                                     書き込みモード。
- |||||
- |||||+----- RWST[リアルタイム・カウンタのウェイト状態フラグ]
- |||||          0 : カウンタ動作中
- |||||          1 : カウンタ値の読み出し, 書き込みモード中
- |||||
- |||||+----- 必ず0に設定

```

```

- |||||
- ||||+----- RIFG[定周期割り込みステータス・フラグ]
- ||||           0 : 定周期割り込み発生なし
- ||||           1 : 定周期割り込み発生あり
- ||||
- |||+----- WAFG[アラーム検出ステータス・フラグ]
- |||           0 : アラーム不一致
- |||           1 : アラームの一致検出
- |||
- ||+----- 必ず0に設定
- ||
- |+----- WALIE[アラーム割り込み(INTRTC)機能の動作制御]
- |           0 : アラームの一致による割り込みを発生しない
- |           1 : アラームの一致による割り込みを発生する
- |
- +----- WALE[アラームの動作制御]
           0 : 一致動作無効
           1 : 一致動作有効

```

\*/

/\*

- リアルタイム・カウンタ・コントロール・レジスタ2

/\*

```

RTCC2 = 0b10000111; /* (RTCC2:リアルタイム・カウンタ・コントロール・レジスタ2)
- |||||
- +|||+++--- RINTE/ICT2-ICT0[インターバル割り込み(INTRTCI)の選択]
- ||||           0xxx : インターバル割り込みを発生しない。
- ||||           1000 : 2^6/fXT (1.953125 ms)
- ||||           1001 : 2^7/fXT (3.90625 ms)
- ||||           1010 : 2^8/fXT (7.8125 ms)
- ||||           1011 : 2^9/fXT (15.625 ms)
- ||||           1100 : 2^10/fXT (31.25 ms)
- ||||           1101 : 2^11/fXT (62.5 ms)
- ||||           111x : 2^12/fXT (125 ms)
- ||||
- ||++----- 必ず0に設定
- ||
- |+----- RCKDIV[RTCDIV端子の出力周波数の選択]
- |           0 : RTCDIV端子から512Hzを出力 (1.95 ms)
- |           1 : RTCDIV端子から16.384kHzを出力(0.061 ms)
- |
- +----- RCLOE2[RTCDIV端子の出力制御]
           0 : RTCDIV端子の出力禁止

```

```

1 : RTCDIV端子の出力許可)

*/

/* RINTE : インターバル割り込み使用 */
/* RCLOE2 : RTCDIV端子の出力禁止 */
/* RCKDIV : 未使用 */
/* ICTn : 周期125ms */

/*****
- 現在時刻初期化 (1/1/1(MON) 00:00:00)
;*****/

/*-----
- 秒カウント・レジスタ
  設定する値は、10進の00-59をBCDコードで設定
----- */
SEC = 0;

/*-----
- 分カウント・レジスタ
  設定する値は、10進の00-59をBCDコードで設定
----- */
MIN = 0;

/*-----
- 時カウント・レジスタ
  設定する値は、10進の00-23 (24時間制)
  または01-12, 21-32 (12時間制) をBCDコードで設定
----- */
HOUR = 0;

WEEK = 0b00000001; /*曜日カウント・レジスタ
- |||||
- |||||+++--- WEEK4/2/1
- ||||| 000 : 日曜日
- ||||| 001 : 月曜日
- ||||| 010 : 火曜日
- ||||| 011 : 水曜日
- ||||| 100 : 木曜日
- ||||| 101 : 金曜日
- ||||| 110 : 土曜日
- |||||
- ++++----- 必ず0に設定

*/

```

```

/*-----
-   日カウント・レジスタ
-   設定する値は，10進の01-31をBCDコードで設定
-----*/
DAY =      0b00000001;          /* 日カウント・レジスタ
-   ||+++++--- 10進の01-31をBCDコードで設定
-   ||
-   +------ 必ず0に設定
*/

/*-----
-   月カウント・レジスタ
-   設定する値は，10進の01-12をBCDコードで設定
-----*/
MONTH =    0b00000001;          /* 月カウント・レジスタ
-   ||||||
-   |||+++++--- MONTH10/MONTH8/MONTH4/MONTH2/MONTH1
-   |||                10進の01-12をBCDコードで設定
-   |||
-   +++----- 必ず0に設定
*/

/*-----
-   年カウント・レジスタ
-   設定する値は，10進の00-99をBCDコードで設定
-----*/
YEAR =     0b00000001;          /* 年カウント・レジスタ
-   ||||||
-   ++++++--- YEAR80/40/20/10/8/4/2/1
-   10進の00-99をBCDコードで設定
*/

RTCE =     1;                  /* RTCE : カウンタ動作開始 */
}

/*****
* Title   : 割込初期設定
*****
* Module  : static void f_ini_itr(void)
* Arg     :
* Ret     :
*****/

```



\* Note :

\*\*\*\*\*/

static void f\_ini\_itr(void)

{

/\*-----

- 割り込み要求フラグ・レジスタ

- 00000000 (IF1H:)

- ||| |||

- ||| |||+--- ADIF (INTAD > 0:割り込み要求無し / 1:割り込み要求中)

- ||| |||+---- RTCIF (INTRTC > 0:割り込み要求無し / 1:割り込み要求中)

- ||| |+----- RTCIIF (INTRTCI > 0:割り込み要求無し / 1:割り込み要求中)

- ||| |+----- KRIF (INTKR > 0:割り込み要求無し / 1:割り込み要求中)

- || |+----- 0

- |+----- 0

- |+----- MDIF (INTMD > 0:割り込み要求無し / 1:割り込み要求中)

- +----- TMIF04 (INTTM04 > 0:割り込み要求無し / 1:割り込み要求中)

-----\*/

RTCIF = 0; /\* RTCIFクリア \*/

RTCIIF = 0; /\* RTCIIFクリア \*/

/\*-----

- 割り込みマスク・フラグ・レジスタ

- 00000000 (MK1H:)

- ||| |||

- ||| |||+--- ADMK (INTAD > 0:割り込み処理許可 / 1:割り込み処理禁止)

- ||| |||+---- RTCMK (INTRTC > 0:割り込み処理許可 / 1:割り込み処理禁止)

- ||| |+----- RTCIMK (INTRTCI > 0:割り込み処理許可 / 1:割り込み処理禁止)

- ||| |+----- KRMK (INTKR > 0:割り込み処理許可 / 1:割り込み処理禁止)

- || |+----- 1

- |+----- 1

- |+----- MDMK (INTMD > 0:割り込み処理許可 / 1:割り込み処理禁止)

- +----- TMMK04 (INTTM04 > 0:割り込み処理許可 / 1:割り込み処理禁止)

-----\*/

RTCMK = 0; /\* INTRTC割り込み許可 \*/

RTCIMK = 0; /\* INTRTCI割り込み許可 \*/

}

/\*\*\*\*\*

\* Title : 割込処理

\*\*\*\*\*

\* Module : \_\_interrupt void INTRTC\_hdr(void)

\* Arg :

```

* Ret   :
* -----
* Note  :
*****/
__interrupt void INTRTC_hdr(void)
{
    if ( RIFG ) {                /* 定周期割込発生 */
        P1 ^=0b00000100;        /* LED2出力を反転する */
        RTCC1 &= 0b11010011;    /* RIFGをクリアする */
    }
    if ( WAFG ) {                /* アラーム割込発生 */
        P1 |= 0b00001000;       /* アラーム発生を表示する */
        salarm();               /* アラーム発生時の処理 */
        RTCC1 &= 0b11001011;    /* WAFGをクリアする */
    }
}

/*****
* Title  : アラーム発生時処理
*****
* Module : void salarm(void)
* Arg    :
* Ret    :
* -----
* Note   : アラーム発生時の処理をここに記載してください。
*****/
void salarm(void){

/*
    ここにアラーム発生時の処理を記載して下さい。
*/

}

/*****
* Title  : 割込処理
*****
* Module : __interrupt void INTRTCI_hdr(void)
* Arg    :
* Ret    :
* -----
* Note   :
*****/

```

```

__interrupt void INTRTCI_hdr(void)
{
    /* インターバル割込発生 */

    P1 ^=0b00000010;          /* P11 (LED1) を反転させる */

}

/*****
* Title   : メイン・ループ
*****
* Module  : void main(void)
* Arg     :
* Ret     :
*-----
* Note    :
*****/
void main(void)
{
    SUBCUD = 0b00000000;      /* 時刻補正無し */

    /*****
    /* SEC ~ YEARカウンタ停止設定      */
    /* カウンタ値読み出し/書き込みモード*/
    *****/
    RWAIT = 1;                /* カウンタ停止設定 */
    while( !RWST );          /* カウンタ停止待ち */

    /*****
    /* 現在時刻設定                      */
    /* 2008/4/10(SUN) 15:59:30          */
    *****/
    SEC   = 0x30;             /* 秒カウント・レジスタを30秒に設定 */
    MIN   = 0x59;             /* 分カウント・レジスタを59分に設定 */
    HOUR  = 0x15;             /* 時カウント・レジスタを15時に設定 */
    WEEK  = 0x00;             /* 曜日カウント・レジスタを日曜日に */
    DAY   = 0x10;             /* 日カウント・レジスタを10日に設定 */
    MONTH = 0x04;             /* 月カウント・レジスタを4月に設定 */
    YEAR  = 0x08;             /* 年カウント・レジスタを08年に設定 */

    /*****
    /* カウンタ動作設定                  */
    *****/

```

```

RWAIT = 0;          /* カウンタ動作設定 */
while( RWST );     /* カウンタ動作待ち */

/*****/
/* アラーム設定          */
/* 毎日 16:00:00        */
/*****/
WALE = 0;          /* 一致動作無効 */

ALARMWM = 0;       /* アラーム分レジスタ:00分 */
ALARMWH = 0x16;    /* アラーム時レジスタ:16時 */
ALARMWW = 0b01111111; /* アラーム週レジスタ:毎日 */
/*
  - |||||+----- WW0[日曜日アラーム設定]
  - |||||          0 : 日曜日アラームOFF設定
  - |||||          1 : 日曜日アラームON設定
  - |||||
  - |||||+----- WW1[月曜日アラーム設定]
  - |||||          0 : 月曜日アラームOFF設定
  - |||||          1 : 月曜日アラームON設定
  - |||||
  - ||||+----- WW2[火曜日アラーム設定]
  - ||||          0 : 火曜日アラームOFF設定
  - ||||          1 : 火曜日アラームON設定
  - ||||
  - |||+----- WW3[水曜日アラーム設定]
  - |||           0 : 水曜日アラームOFF設定
  - |||           1 : 水曜日アラームON設定
  - |||
  - ||+----- WW4[木曜日アラーム設定]
  - ||            0 : 木曜日アラームOFF設定
  - ||            1 : 木曜日アラームON設定
  - ||
  - |+----- WW5[金曜日アラーム設定]
  - |             0 : 金曜日アラームOFF設定
  - |             1 : 金曜日アラームON設定
  - |
  - |+----- WW6[土曜日アラーム設定]
  - |             0 : 土曜日アラームOFF設定
  - |             1 : 土曜日アラームON設定
  - |
  - +----- 必ず0に設定
*/

```

```
WALIE = 1;          /* アラーム割込有効設定 */
WALE  = 1;          /* 一致動作有効 */

EI();

while(1) {
    HALT();
    NOP();
}
}
```

## 付録B 改版履歴

版 数	発行年月	改版箇所	改版内容
第1版	-	-	-

---

## 【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

---

### お問い合わせ先

#### 【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL（アドレス） <http://www.necel.co.jp/>

---

#### 【営業関係，デバイスの技術関係お問い合わせ先】

半導体ホットライン

（電話：午前 9:00～12:00，午後 1:00～5:00）

#### 【マイコン開発ツールの技術関係お問い合わせ先】

開発ツールサポートセンター

電 話：044-435-9494

E-mail：[info@necel.com](mailto:info@necel.com)

E-mail：[toolsupport-micom@ml.necel.com](mailto:toolsupport-micom@ml.necel.com)

---

#### 【資料請求先】

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

---