

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。

M16C/62P グループ

発振停止、再発振検出機能の応用例

1. 要約

この資料は、M16C/62P グループの発振停止、再発振検出機能の応用例を掲載しています。サンプルプログラムは、メインクロックが停止した場合でもオンチップオシレータによりプログラムを継続動作させる例を説明しています。

2. はじめに

この資料で説明する応用例は、次のマイコン、条件での利用に適用されます。

| | |
|-----------|---------------------------------------|
| マイコン | M16C/62P グループ |
| 評価 / 動作環境 | Starter-Kit for M16C/62P (M3A-0664) |
| メインクロック | 6MHz |
| CPU クロック | PLL クロック 24MHz (メインクロックの 4 遁倍) |

M16C/62P グループと同様の SFR(周辺機能制御レジスタ)を持つ他の M16C ファミリでも本プログラムを使用することができます。ただし、一部の機能を機能追加等で変更している場合がありますのでマニュアルで確認してください。このアプリケーションノートをご使用に際しては十分な評価を行ってください。

3. 機能概要

本サンプルプログラムで使用している機能について概要を説明します。

各機能の詳細については M16C/62P グループハードウェアマニュアルを参照ください。

| 機能 | 詳細についての参照先 |
|--------------------|---------------------------|
| 3.1.オンチップオシレータクロック | 「クロック発生回路」 |
| 3.2.発振停止、再発振検出機能 | 「クロック発生回路」、「発振停止、再発振検出機能」 |

本サンプルプログラムで設定している関連レジスタを「Appendix A」に添付します。詳細は最新のハードウェアマニュアルを参照ください。

3.1. オンチップオシレータクロック

CPU が内蔵しているオンチップオシレータが供給する約 1MHz のクロックです。

CPU クロックと周辺機能クロックのクロック源になります。

リセット後、オンチップオシレータは停止しています。

(オンチップオシレータの仕様はハードウェアマニュアルの「電気的特性」を参照してください)

3.2. 発振停止、再発振検出機能

発振停止、再発振検出機能は外部要因によりメインクロックが停止した場合に備えた機能です。
発振停止、再発振検出機能は、メインクロック発振回路の停止と再発振を検出します。

発振停止、再発振検出機能は CM2 レジスタの CM20 ビットで、有効、無効が選択できます。

発振停止、再発振検出時にはリセットまたは発振停止、再発振検出割り込みを発生します。
どちらを発生させるかは、CM2 レジスタの CM27 ビットで選択できます。

表 3.1 に発振停止、再発振検出機能の仕様をまとめます。

表 3.1 発振停止、再発振検出機能の仕様

| 項目 | 仕様 |
|-------------------|--|
| 発振停止検出可能クロックと周波数域 | f(XIN) 2MHz |
| 発振停止、再発振検出機能 有効条件 | CM20 ビットを"1"(有効)にする |
| 発振停止、再発振検出時の動作 | <ul style="list-style-type: none"> ・ CM27 ビット=0 のとき : リセット発生 ・ CM27 ビット=1 のとき : 発振停止、再発振検出割り込み発生 |

4. サンプルプログラム仕様

Starter-Kit for M16C/62P (M3A-0664) を使用

メインクロック 6MHz

通常は、CPU クロック源を PLL 4 通倍 (24MHz) クロックでプログラムを動作させる。

発振停止、再発振検出割り込み機能を有効にする。

メインクロックの発振停止を検出した場合は、CPU クロック源をオンチップオシレータに切り替える。

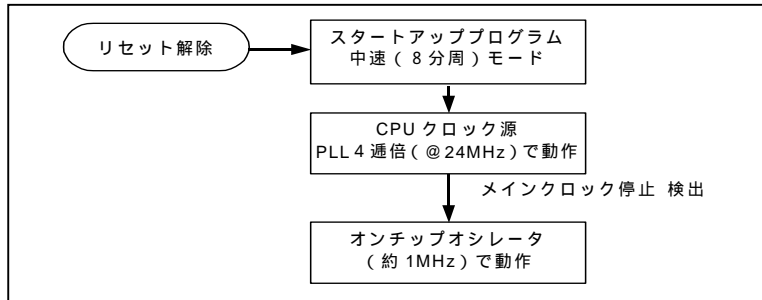


図 4.1 サンプルプログラム動作概要

サンプルプログラム仕様

CPU クロック源が PLL 4 通倍 (24MHz) の時は 125ms 間隔で LED の点灯を繰り返す。

24MHz の CPU クロック源で 1ms 周期をタイマ A0 で生成し、125 回カウントすることで、125ms 間隔を作る。

LED に使用するポートはスタータキットに準じて P0_7~P0_0、P1_0,P1_1 を使用する。

以下に Starter Kit for M16C/62P (M3A-0664) のポート接続を示します。

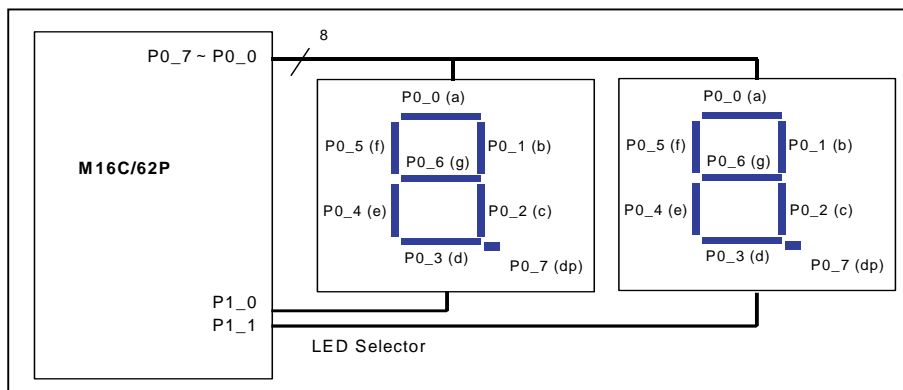


図 4.2 Starter Kit for M16C/62P (M3A-0664) のポート接続

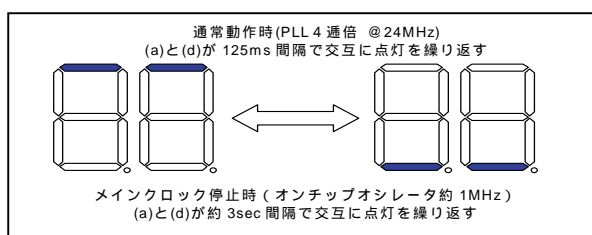
サンプルプログラムの動作

(1)通常動作時は CPU クロック源を PLL 24MHz として LED の(a)と(d)が交互に点灯を繰り返す。

LED の点灯間隔は 125ms。

(2)メインクロックが発振停止すると CPU クロック源はオンチップオシレータ (約 1MHz) に切り替わる。

LED の点灯間隔は $125\text{ms} * 24 = 3\text{sec}$ となりプログラム動作を継続する。

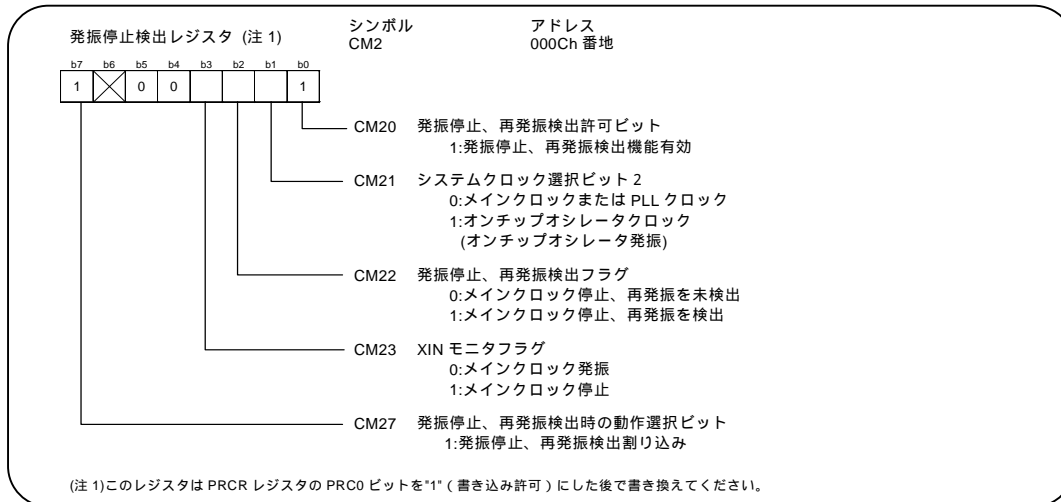


4.1. レジスタの設定

サンプルプログラムで使用している機能に関連したレジスタの設定について説明します。

(1) CM2 (発振停止検出レジスタ) の設定

- ・ CM20 ビットを"1"にして、発振停止、再発振検出機能を"有効"にします。
- ・ CM27 ビットを"1"にして、発振停止、再発振検出時の動作を"発振停止、再発振検出割り込み発生"に設定します。



(2) 固定ベクタテーブルの設定

固定ベクタテーブルのベクタ番地 0FFFF0h ~ 0FFFF3h に発振停止、再発振検出機能の割り込み処理のアドレスを設定します。固定ベクタテーブルについては、M16C/62P グループハードウェアマニュアルの「割り込み」を、具体的な設定方法については「6.1 sect30.inc (セクション定義ファイル)」とサンプルプログラムを参照ください。

(3) 発振停止、再発振検出割り込み発生時の処理

メインクロックの停止または再発振を検出すると発振停止、再発振検出割り込みがかかり、CM22 ビットが"1"になります。このとき、発振停止、再発振検出割り込みは禁止されます。CPU クロック源を PLL クロックにしている場合、CM21 ビットは変化しないので、割り込み処理内で"1"にして、オンチップオシレータを CPU クロック源にします。

(4) メインクロックの状態判定

発振停止、再発振検出割り込み処理内で CM23 ビットを数回読むことにより、メインクロックが発振しているか停止しているかを判定します。

5. サンプルプログラムフローチャート

5.1. 概要フロー

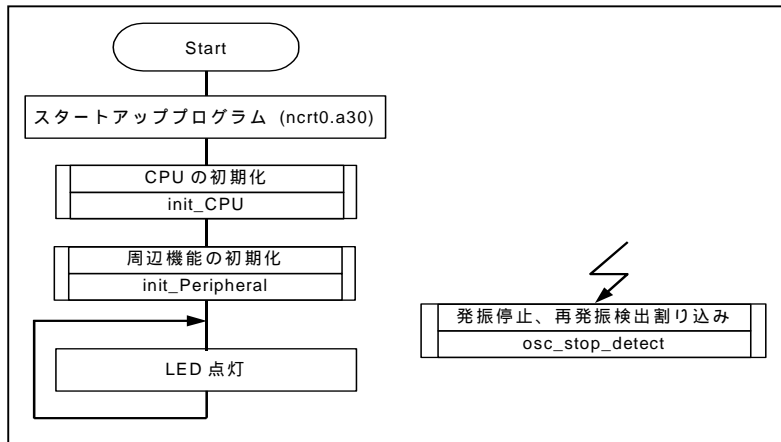


図 5.1 サンプルプログラム概要フロー図

5.2. CPUの初期化フロー (init_CPU)

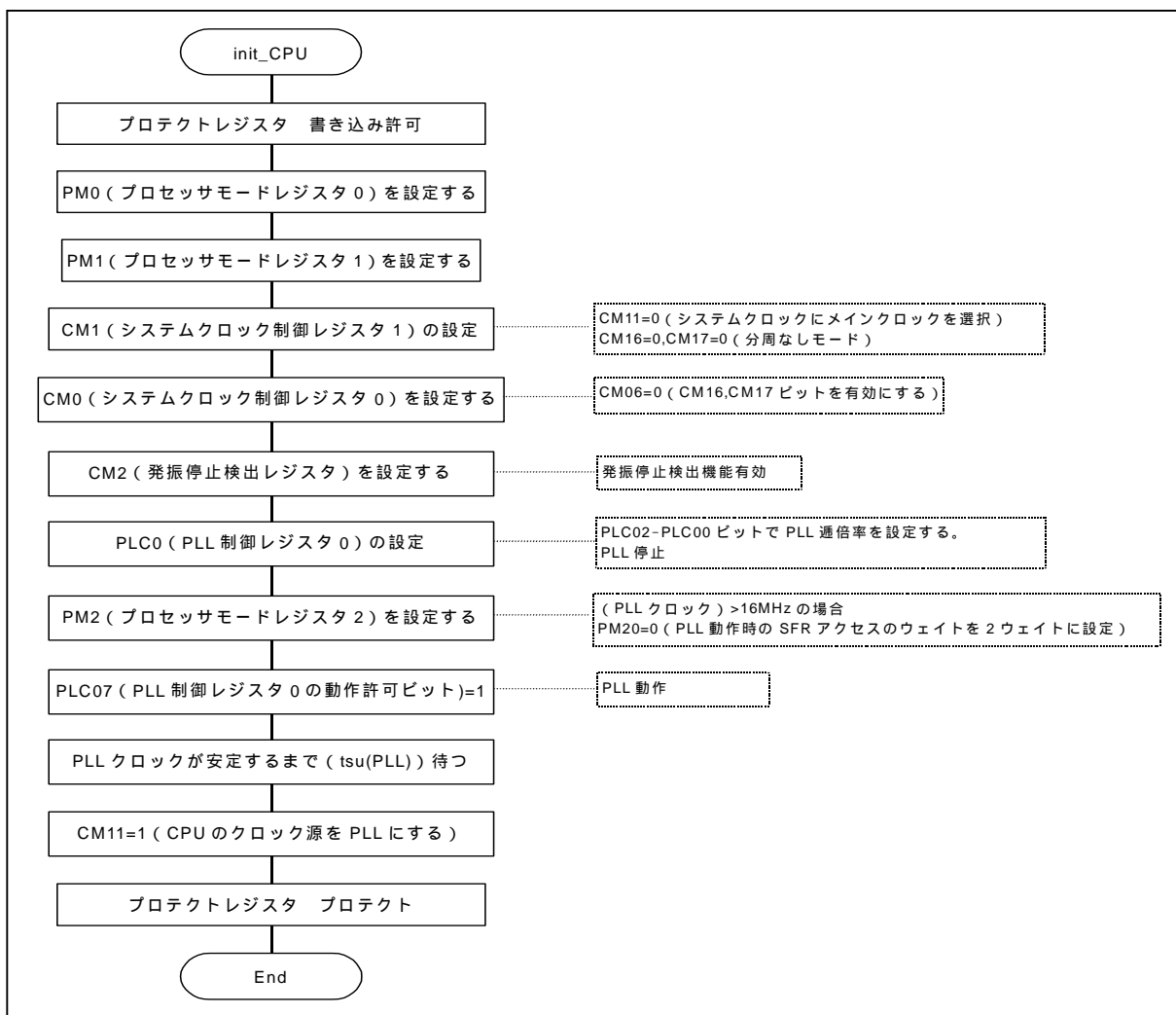


図 5.2 CPUの初期化フロー図

5.3. 周辺機能の初期化フロー (init_Peripheral)

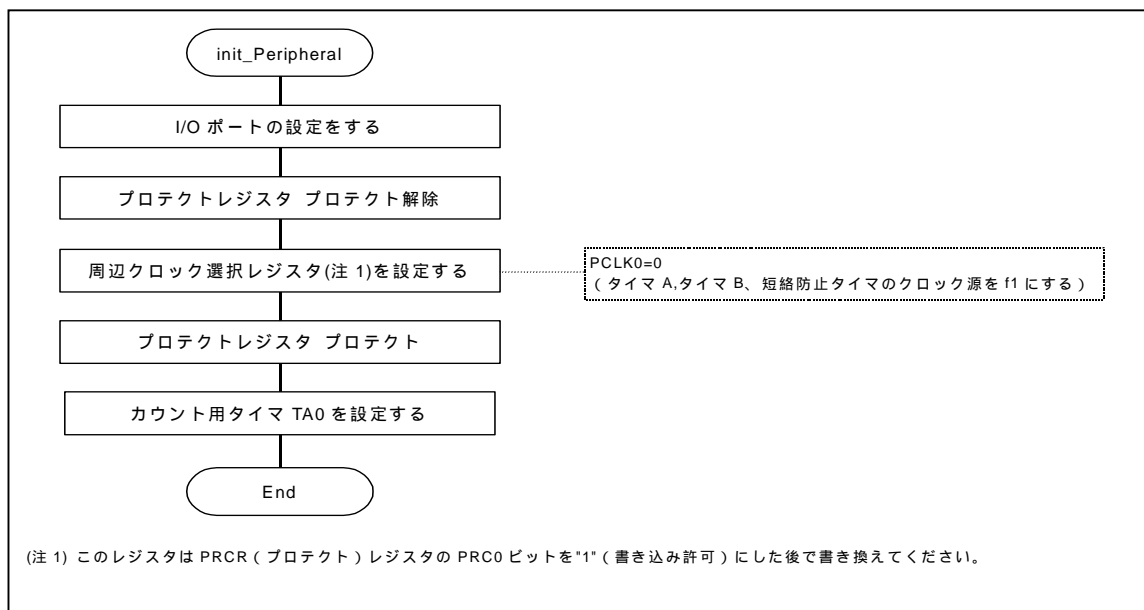


図 5.3 周辺機能の初期化フロー図

5.4. 発振停止、再発振検出割り込み処理フロー (osc_stop_detect)

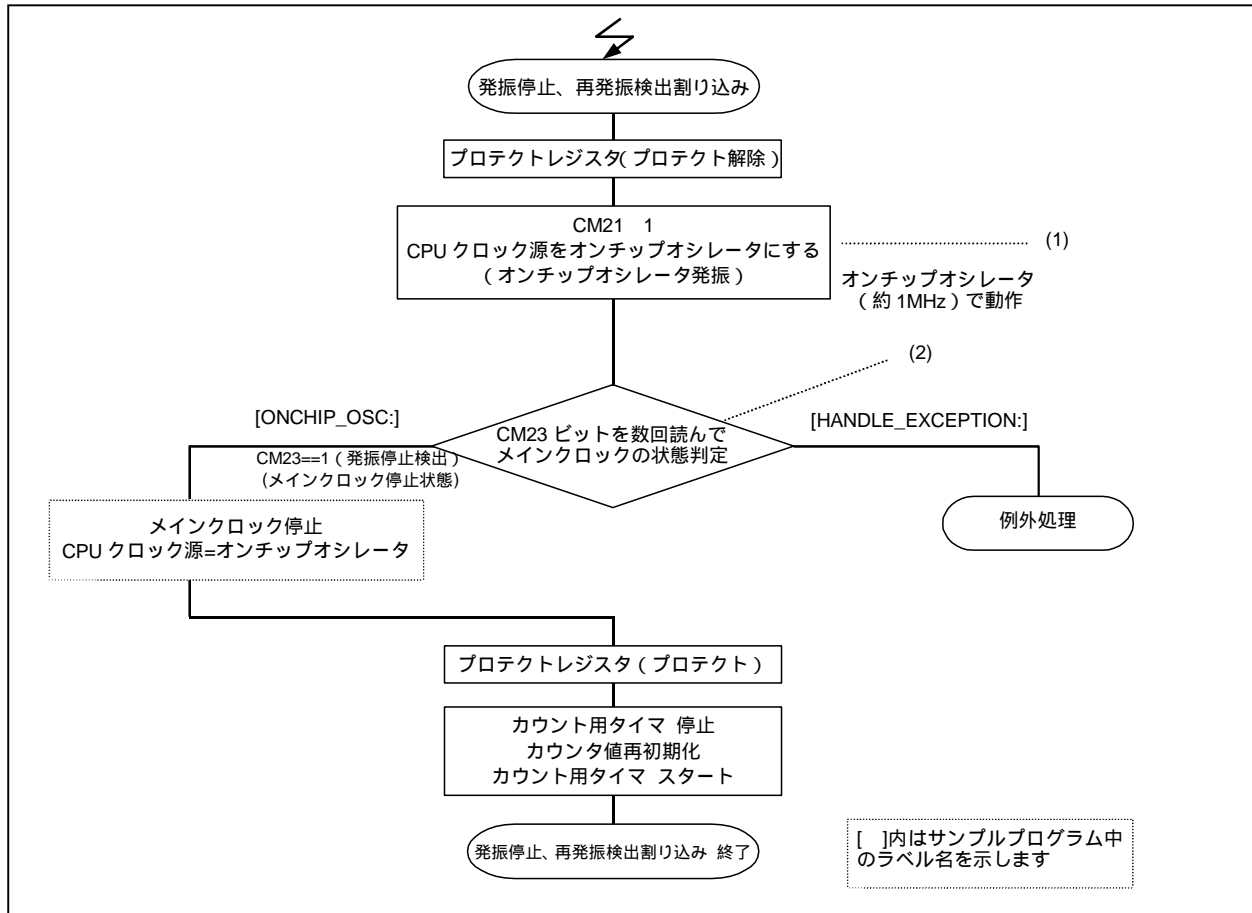


図 5.4 発振停止、再発振検出割り込み処理フロー図

【説明】

- (1) サンプルプログラムは、CPU クロック源として PLL クロックを選択しているため、CM21 ビットは変化しません。そのため、割り込み処理先頭で"1" (オンチップオシレータクロック) にしています。
- (2) メインクロックの状態判定は、割り込み処理内で CM23 ビット (XIN モニタフラグ) を数回読んで判定しています。

サンプルプログラムでは、CM23 ビットの値が連続して数回"1"に安定すればメインクロック発振停止と判定しています。CM23 ビットの値が連続して"1"に安定しなかった場合は、例外として扱っています。CM23 ビットのチェック方法の例については次頁を参照ください。

5.5. CM23 ビット (XIN モニタフラグ) のチェック例

サンプルプログラムでは、メインクロックの状態判定を CM23 ビット (XIN モニタフラグ) を数回読むことで判定しています。サンプルプログラムのチェックフローを以下に示します。

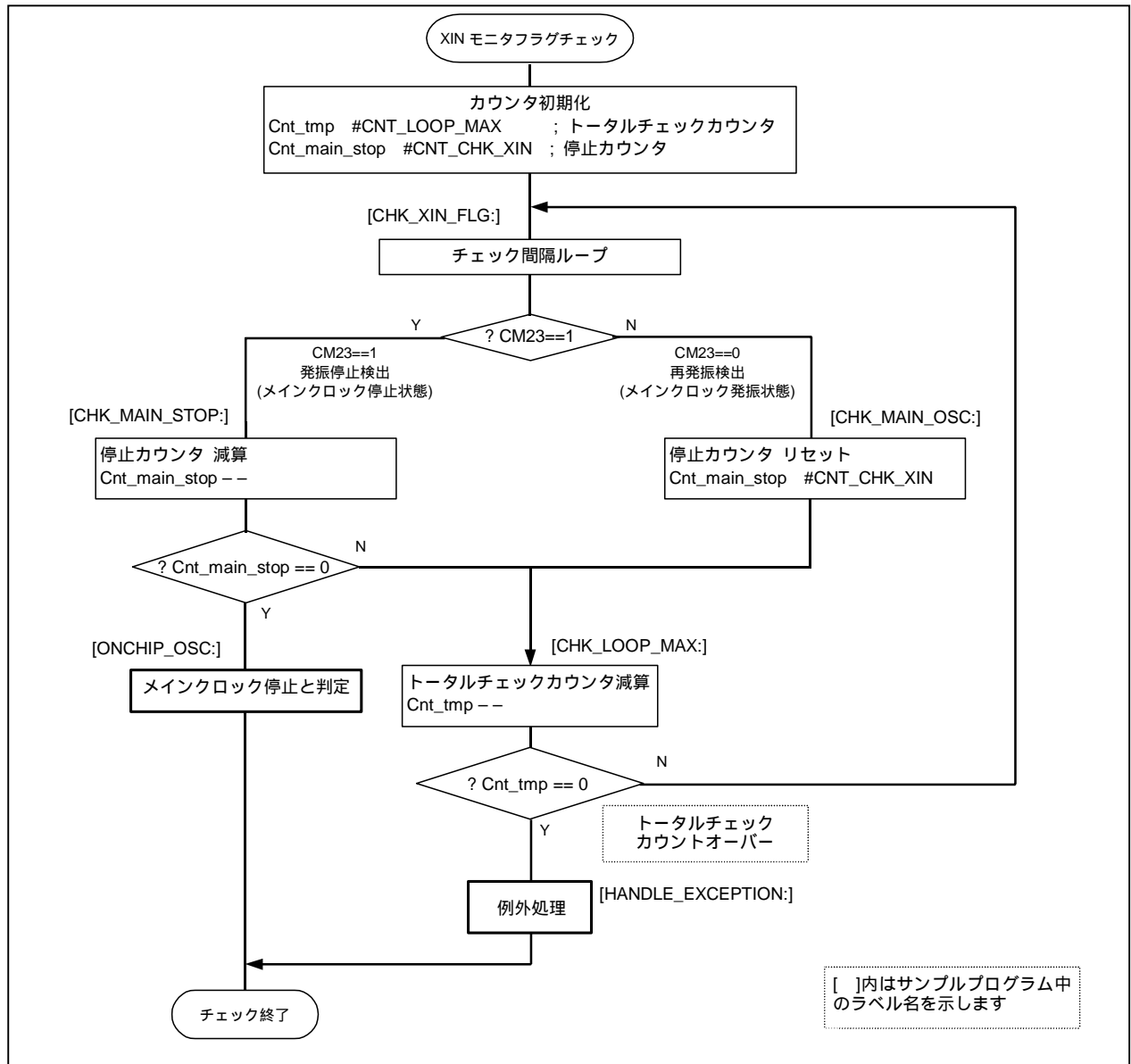


図 5.5 CM23 ビット (XIN モニタフラグ) のチェックフロー図

【説明】

サンプルプログラムでは、CM23 ビットを間隔をあけてチェックし、CM23 ビットが連続して CNT_CHK_XIN(回)"1"に安定した場合にメインクロック停止と判定しています。

以下にサンプルプログラムの CM23 ビットの判定例について説明します。

5.5.1 CM23 ビット (XIN モニタフラグ) の判定例

- (1) CM23 ビットをチェック間隔(約 5ms)でチェックしています。
(チェック間隔はループ処理で作っています。「6.3.A_interrupt.a30」を参照ください。)
- (2) CM23 ビットが連続 CNT_CHK_XIN=8(回)"1"になればメインクロック停止と判定する。
- (3) CM23 ビットのチェック回数が CNT_LOOP_MAX=16(回)を超えれば例外発生と判定する。

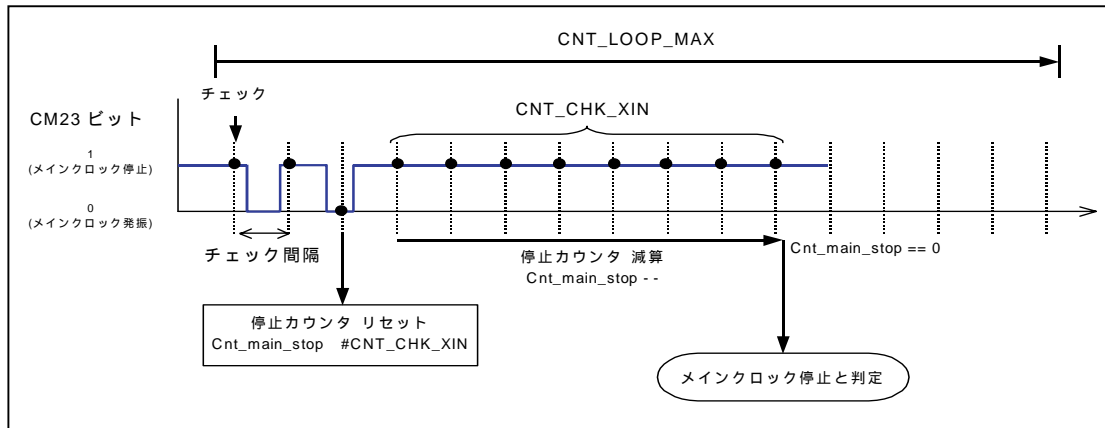


図 5.6 CM23 ビットの判定例

5.6. HANDLE_EXCEPTION (例外処理)

サンプルプログラムでは、発振停止、再発振検出割り込み処理内で CM23 ビット (XIN モニタフラグ) が CNT_CHK_XIN(回)連続して"1" (メインクロック停止状態) で安定しなければ、例外として扱っています。

サンプルプログラムの例外処理は、CPU クロック源をオンチップオシレータに切り替える処理 (ONCHIP_OSC) にジャンプさせて、例外が発生してもオンチップクロックで動作するようにしています。

6. サンプルプログラム

本サンプルプログラムは「4. サンプルプログラム仕様」で示した条件でのサンプルプログラムであり、任意のシステム/アプリケーションにおいて動作を保証するものではありません。

本サンプルプログラムはC言語とアセンブリ言語で記述しています。
また、プログラムのROM化にはCコンパイラ NC30WA Ver.5.30 Release 1 に付属しているサンプルのスタートアッププログラムとセクション定義ファイルを流用しています。

使用コンパイラ

- ・ NC30WA Ver.5.30 Release 1
- ・ サンプルプログラムのコンパイルオプション (詳細は NC30 のユーザーズマニュアルを参照ください)

| | |
|-----------------|--|
| -c | リロケータブルファイル(拡張子.r30)を作成し、処理を終了します。 |
| -dSL | C言語ソースリストをコメントとして出力したアセンブリ言語ソースファイル(拡張子".a30")を生成します。さらにアセンブリ言語リストファイル(拡張子".lst")を生成します。 |
| -fER | レジスタ記憶クラスを有効にします。 |
| -g | デバッグ情報をアセンブリ言語ソースファイル(拡張子".a30")に出力します。これにより、C言語レベルデバッグが可能になります。 |
| -finfo | インスペクタ、"Stk Viewer"、"Map Viewer"、"utI30"、に必要な情報を出力します。エントリー版では、本オプションは指定できません。 |
| -Wall | 検出可能な警告をすべて表示します。 |
| -WEF エラー出力ファイル名 | エラーメッセージを指定したファイルに出力します。 |

以下にファイル構成について説明します。

ファイル構成

| ファイル名 | 内 容 | |
|---------------------------|---|--|
| sfr62p.h, sfr62p.inc | M16C/62P 用の SFR 定義ファイルです。最新のファイルを入手してご使用ください。 | |
| ncrt0.a30 | スタートアッププログラム (NC30WA Ver.5.30 Release 1 に添付されているファイルと同じです) | |
| sect30.inc(セクション定義ファイル) | セクション定義ファイル (NC30WA Ver.5.30 Release 1 に添付されているファイルを流用しています) ・ 固定割り込みベクタに割り込み処理プログラムを登録しています ・ プログラムサイズが小さいのでプログラムの配置アドレスを変更しています。 | |
| board.h | サンプルプログラム用定義ファイル (Starter Kit for M16C/62P M3A-0664 用) | |
| C_main.c | メインプログラム | |
| | 関数名 | 処理概要 |
| | main | スタートアッププログラムから Call されるメイン関数です。 無限ループで LED の点滅処理を行います。 |
| | init_CPU | マイコンの初期設定を行います。 マイコンの動作モード、発振停止検出レジスタなどの初期設定を行います。 |
| A_interrupt.a30 | 割り込み処理プログラム | |
| | 関数名 | 処理概要 |
| | osc_stop_detect | 発振停止、再発振検出割り込み処理です。 発振停止、再発振検出割り込み要求が発生すると、この処理が実行されます。 |

6.1. sect30.inc (セクション定義ファイル)

セクション定義ファイルでは、Flash ROM エリアの先頭アドレスの変更と、
発振停止、再発振検出割り込み処理(osc_stop_detect)を固定ベクタテーブルに登録しています。

```

*****
;
; C Compiler for R8C/Tiny, M16C/60,30,20,10
; COPYRIGHT(C) 1999(2000-2002) RENESAS TECHNOLOGY CORPORATION
; AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED
;
;
; Written by T.Aoyama
;
; sect30.inc      : section definition
; This program is applicable when using the basic I/O library
;
; $Id: sect30.inc,v 1.22 2004/02/16 05:17:14 muranaka Exp $
;
; 2005.02.25 Ver.1.00
; Sample operation of Oscillation Stop and Re-ocsillation Detect Function.
; This file was diverted from the sect30.inc attached to
; the NC30WA Ver.5.30 Release 1.
; The lines changed from original source are marked by @100
*****

.if __R8C__ != 1
;
; for M16C/60,30,20,10
;
;-----
; Arrangement of section
;-----
; Near RAM data area
;-----
; SBDATA area
; .section data_SE,DATA
; .org 400H
data_SE_top:

; .section bss_SE,DATA,ALIGN
bss_SE_top:

; .section data_SO,DATA
data_SO_top:

; .section bss_SO,DATA
bss_SO_top:

; near RAM area
; .section data_NE,DATA,ALIGN
data_NE_top:

; .section bss_NE,DATA,ALIGN
bss_NE_top:

; .section data_NO,DATA
data_NO_top:

; .section bss_NO,DATA
bss_NO_top:

;-----
; Stack area
;-----
; .section stack,DATA
; .b1kbSTACKSIZE
stack_top:

; .b1kbISTACKSIZE
istack_top:

```

```

-----
; heap section
-----
.section heap,DATA
heap_top:
    .b1kbHEAPSIZE
-----
; Near ROM data area
-----
.section rom_NE,ROMDATA,ALIGN
rom_NE_top:

.section rom_NO,ROMDATA
rom_NO_top:
-----
; Far RAM data area
-----
.section data_FE,DATA
.org 10000H
data_FE_top:

.section bss_FE,DATA,ALIGN
bss_FE_top:

.section data_FO,DATA
data_FO_top:

.section bss_FO,DATA
bss_FO_top:
-----
; Far ROM data area
-----
.section rom_FE,ROMDATA
.if 0
.org 0F0000H
.ELSE
.org 0FF000H ; @100 Use Flash memory area from Block 0
.ENDIF
rom_FE_top:

.section rom_FO,ROMDATA
rom_FO_top:

( 中略 )
:
:

```

プログラムサイズが小さいので Flash ROM エリアを Block0 から使用するよう
に変更しています

```

;=====
; fixed vector section
;-----
        .section fvector,ROMDATA
;        .org 0ffffdch
;UDI:
;        .lword    dummy_int
;OVER_FLOW:
;        .lword    dummy_int
;BRKI:
;        .lword    dummy_int
;ADDRESS_MATCH:
;        .lword    dummy_int
;SINGLE_STEP:
;        .lword    dummy_int

```

発振停止、再発振検出割り込み処理 (ocs_stop_detect) を
固定ベクタテーブルに登録しています。

```

; @100
; Interrupt handling routine of Oscillation Stop and Re-oscillation Detect Function
.glb osc_stop_detect
        .org 0ffff0h          ; Vector Table Address of
WDT:    .lword    osc_stop_detect; WDT, Oscillation Stop and, Re-oscillation, Voltage Down Detection

```

```

;DBC:
;        .lword    dummy_int
;NMI:
;        .lword    dummy_int
;        .org 0ffffch
RESET:
        .lword    start

.else; __R8C__
(以下省略)
:
:

```

6.2. C_main.c

サンプルプログラムを動作させる場合、以下の値はコンパイルオプションやシステムに応じて設定値を変更してください。

| 定義 | 内容 |
|--------------|--|
| CNT_WAIT_PLL | 関数 init_CPU で使用しています。 PLL 発振安定待ち時間 tsu(PLL)を作るためのループ回数を定義しています。 なお、正確に tsu(PLL)以上の待ち時間を作るにはタイマを利用することを推奨します。 |

```

/*****
*
* FILE      : C_main.c
* CONTENTS  : Sample operation of Oscillation Stop and Re-oscillation Detect Function
*            Main routine
*
* CPU       : M16C/62P (Starter Kit for M16C/62P :M3A-0644)
* OS        : No used
* COMMENT   :
* HISTORY   : 2005.02.25 Ver.1.00
*
* NOTE      :
*
* Copyright(C)2004, Renesas Technology Corp.
* Copyright(C)2004, Renesas Solutions Corp.
* All rights reserved.
*
*****/
*/

/*****
*
* Include files
*****/
*/
#include "sfr62p.h"
#include "board.h"

/*****
*
* Defines
*****/
*/
/* Set the value suitable for the system */
#define CNT_WAIT_PLL    (14000) /* Count value of
                                the PLL Frequency Synthesizer Stabilization Wait Time (@6MHz) */

#define MS_INTERVAL    (125) /* Count up by 125ms @24MHz PLL */
#define TAO_INTERVAL   (24000-1) /* 1ms @24MHz PLL, f1 */
#define INIT_LED_data  ( ~(_a_) )
#define MASK_FLICKER   ( _a_ | _d_ )

/*****
*
* Global Variables
*****/
*/
unsigned short Cnt_ms;

/*****
*
* Prototypes
*****/
*/
void main( void );
void init_CPU( void );
void init_Peripheral( void );

```



```

/*****
MODULE : main
FUNCTION: Main routine of sample program
PARAMETERS: None
RETURN : None
*****/
*/
void main( void )
{
    init_CPU();
    init_Peripheral();

    Cnt_ms = 0;
    LED_PORT = INIT_LED_data;

    ta0s = 1; // Start Period count Timer

    while(1){
        // ? TAO's interrupt request is generated
        if( ir_ta0ic == 1 ){
            ta0ic = 0; // Clear Interrupt Request Bit
            if ( MS_INTERVAL <= (Cnt_ms++) ){
                Cnt_ms = 0;
                LED_PORT ^= MASK_FLICKER;
            }
        }
    }
}

```

```

/*****
MODULE : init_CPU
FUNCTION: Initialize CPU
        Processor Mode: Single-Chip Mode
        CPU Clock: PLL Clock (Xin = 6MHz, Multiply by 4 --> 24MHz)
        Oscillation Stop, Re-Oscillation Detection function enabled
PARAMETERS: None
RETURN    : None
*****/
*/
void init_CPU( void )
{
    volatile short cnt_word;

    prcr = 0x03;    // Protect Register (write enabled)

    //=====
    // Setting Processor Mode
    //=====
    // Processor Mode Register 0
    pm0 = 0x00;
    // (b1-b0) Processor Mode Bit           [00:Single-Chip]
    // (b2) R/W Mode Select Bit            [0:(Note2)]
    // (b3) Software Reset Bit             [0]
    // (b5-b4) Multiplexed Bus Space Select Bit [00:(Note2)]
    // (b6) Port P4_0 to P4_3 Function Select Bit [0:(Note2)]
    // (b7) BCLK Output Disable Bit        [0:(Note2)]
    // (Note2) Effective when memory expansion mode or microprocessor mode

    // Processor Mode Register 1
    pm1 = 0x08;
    // (b0) CS2 Area Switch Bit             [0:Block A disable]
    // (b1) Port P3_7 to P3_4 Function Select Bit [0:(Note3)]
    // (b2) Watchdog Timer Function Select Bit [0:Watchdog timer interrupt]
    // (b3) Internal Reserved Area Expansion Bit [1:The entire area is usable]
    // (b5-b4) Memory Area Expansion Bit      [00:(Note3)]
    // (b6) Reserved bit (Set to "0")        [0]
    // (b7) Wait Bit                        [0:No wait state]
    // (Note3) Effective when memory expansion mode or microprocessor mode

    //=====
    // Using the PLL clock as the clock source for the CPU
    //=====
    // System Clock Control Register 1
    cm1 = 0x20;
    // (b0) All Clock Stop Control Bit       [0:Clock on]
    // (b1) System Clock Select Bit 1        [0:Main clock] (Note5)
    // (b4-b2) Reserved Bit (Set to "0")     [000]
    // (b5) XIN-XOUT Drive Capacity Select Bit [1:HIGH]
    // (b7-b6) Main Clock Division Select Bit [00:No division mode]
    // (Note5) After setting the PLC07 bit in the PLC0 register to "1"(PLL operation),
    //          wait until Tsu(PLL) elapses before setting the CM11 bit to "1"(PLL Clock)

    // System Clock Control Register 0
    cm0 = 0x00;
    // (b1-b0) Clock Output Function Select Bit [00: I/O port P5_7]
    // (b2) WAIT Mode Peripheral Function Clock Stop Bit
    //          [0:Do not stop peripheral function clock in wait mode]
    // (b3) XCIN-XCOUT Drive Capacity Select Bit [0:Low]
    // (b4) Port XC Select Bit                 [0:I/O port P8_6, P8_7]
    // (b5) Main Clock Stop Bit               [0:0n]
    // (b6) Main Clock Division Select Bit     [0:CM16 and CM17 valid]
    // (b7) System Clock Select Bit           [0:Main clock, PLL clock, or on-chip oscillator clock]

```

```

//=====
// Enable Detection Stop, Re-oscillation Detection Function
//=====
// Oscillation Stop Detection Register
cm2 = 0x81;
// (b0) Oscillation Stop, Re-Oscillation Detection Bit
// [1:Detection function enabled]
// (b1) System Clock Select Bit 2 [0:Main clock or PLL clock]
// (b2) Oscillation Stop, Re-Oscillation Detection Flag [0:(Note4)]
// (b3) XIN Monitor Flag [0:(Read Only)]
// (b5-b4) Reserved Bit (Set to "0") [00]
// (b6) Nothing is assigned (When write, set to "0") [0]
// (b7) Operation Select Bit [1:Oscillation stop, re-oscillation detection interrupt]
// (Note4) This flag is set to "1" when the main clock is detected to have stopped and
// the main clock is detected to have restarted oscillating.

// PLL Control Register 0 (Set the PLC02 to PLC00 bits (multiplying factor))
plc0 = 0x12;
// (b2-b0) PLL Multipling Factor Select Bit [010:Multiply by 4]
// (b3) Nothing is assigned (When write, set to "0")
// [0]
// (b4) Reserved Bit (Set to "1") [1]
// (b6-b5) Reserved Bit (Set to "0") [00]
// (b7) Operation Enable Bit [0:PLL Off]

// Processor Mode Register 2 (Set the PM20 bit to "0" (2 wait states))
pm2 = 0x00;
// (b0) Specifying Wait when Accessing SFR at PLL Operation
// [0:2 waits (Note2)]
// (b1) System Clock Protective Bit [0:Clock is protected by PRCR register]
// (b2) WDT Count Source Protective Bit [0:CPU clock is used for the WDT count source]
// (b4-b3) Reserved Bit (Set to "0") [00]
// (b7-b5) Nothing is assigned (When write, set to "0")
// [000]
// (Note2) The PM20 bit become effective when PLC07 bit in the PLC0 register is
// set to "1"(PLL On).
// Change the PM20 bit when the PLC07 bit is set to "0"(PLL Off).
// Set the PM20 bit to "0"(2 waits) when PLL clock > 16MHz.

// Set the PLC07 bit to "1" (PLL On)
plc07 = 1;
// (b7) Operation Enable Bit [1:PLL On]

// Wait until the PLL clock becomes stable (tsu(PLL))
cnt_word = CNT_WAIT_PLL; while( cnt_word -- );

// Set the CM11 bit to "1" (PLL clock for the CPU clock source)
cm11 = 1;
// (b1) System Clock Select Bit 1 [1:PLL clock]

prcr = 0x00; // Protect Register (write protected)
}

```

```

/*****
MODULE : init_Peripheral
FUNCTION: Initialize Peripheral
PARAMETERS: None
RETURN : None
*****/
*/
void init_Peripheral( void )
{
//=====
// Setting I/O port
//=====
LED_PORT = LED_data_blank; // Initialize LED Port
ASSERT_LEDS; // Assert tens and first digit LED
LED_PD = 0xff; // LED Port Direction
LED_SELECTOR_PD |= 0x03; // LED Selector Port Direction

//=====
// Peripheral Clock Select Register
//=====
prcr = 0x01; // Protect Register (write enabled)
// Peripheral Clock Select Register
pclkr = 0x03;
// (b0) Timers A, B Clock Select Bit [1:f1]
// (b1) S1/O Clock Select Bit [1:f1S1O]
// (b7-b2) Reserved bit (Set to "0") [000000]
prcr = 0x00; // Protect Register (write protected)

//=====
// Timer A0 (Timer Mode, 1ms)
//=====
// Stop counting
ta0s = 0;
// Timer Mode Register
ta0mr = 0x00;
// (b1-b0) Operation Mode Select Bit [00:Timer mode]
// (b2) Pulse Output Function Select Bit [0:Pulse is not output]
// (b4-b3) Gate Function Select Bit [00:Gate function not available]
// (b5) Set to "0" in timer mode [0]
// (b7-b6) Count Source Select Bit [00:f1 or f2]
// Interrupt Priority Level (Interrupt disable)
ta0ic=0;
// Timer Register
ta0 = TAO_INTERVAL;
}

```

6.3. A_interrupt.a30

サンプルプログラムの発振停止、再発振検出割り込み処理を以下に示します。
サンプルプログラムを動作させる場合、以下の値は必要に応じて設定してください。
発振停止検出ロジックはシステムや発振子に合わせて変更してください。

| 定義 | 内容 |
|--------------|---|
| CNT_CHK_XIN | CM23 ビット(XIN モニタフラグ)カウント回数 CM23 ビットが連続 CNT_CHK_XIN(回)"1"に安定した場合、メインクロック停止と判定します。 |
| CNT_LOOP_MAX | トータルチェック回数 CM23 ビットチェック回数の上限値 |
| CNT_WAIT_CHK | チェック間隔 (チェック間隔は CNT_WAIT_CHK(回)ループするループ処理で作っています) |

```

;*****
;
;
; FILE      : A_interrupt.a30
; CONTENTS  : Sample operation of Oscillation Stop and Re-oscillation Detect Function
;             Interrupt handling routine
;
; CPU       : M16C/62P (Starter Kit for M16C/62P :M3A-0644)
; OS        : No used
; COMMENT   :
; HISTORY   : 2005.02.25 Ver.1.00
;
; NOTE      :
;
; Copyright(C)2004, Renesas Technology Corp.
; Copyright(C)2004, Renesas Solutions Corp.
; All rights reserved.
;*****
;
;*****
; Include files
;*****
; .list OFF
; .include sfr62p.inc
; .list ON
;*****
; Defines
;*****
; Set the values suitable for the system
CNT_CHK_XIN .equ (8) ; Number of Oscillation stop counting
CNT_LOOP_MAX .equ (16) ; Number of total check
CNT_WAIT_CHK .equ (500) ; Number of loop that make check interval (Approx. 5ms @1MHz)

TAO_INTERVAL .equ (24000-1) ; 1ms @24MHz PLL, f1
;*****
; Global symbols
;*****
; .glb osc_stop_detect
; .glb _Cnt_ms
;*****
; Variables
;*****
; .section bss_NE, DATA
Cnt_main_stop: .blkw 1
Cnt_tmp: .blkw 1
Cnt_wait: .blkw 1

```

```

.section    program, align
;*****
;
; MODULE : osc_stop_detect
; FUNCTION: Interrupt handling routine of
;           Oscillation Stop and Re-oscillation Detect Function.
;
; NOTE   : This routine does not use any registers.
;           Except for automatically saved registers, if there are any registers
;           that are likely to be modified in the interrupt handling routine,
;           save them to the stack in software.
;
; PARAMETERS: None
; RETURN   : None
;*****
osc_stop_detect:
    mov.b #01h, prcr          ; Protect Register (write enabled)
    bset  cm21                ; On-chip oscillator oscillating
                                ; (Where the PLL clock corresponds to the CPU clock source,
                                ; the CM21 bit remains unchanged)
;=====
; Check XIN Monitor Flag
;=====
    mov.w #CNT_LOOP_MAX, Cnt_tmp
    mov.w #CNT_CHK_XIN, Cnt_main_stop

CHK_XIN_FLG:
    mov.w #CNT_WAIT_CHK, Cnt_wait
WAIT_CHK:
    adjnz.w #-1, Cnt_wait, WAIT_CHK

    ; Check CM23 (XIN Monitor Flag)
    btst  cm23
    jz    CHK_MAIN_OSC

CHK_MAIN_STOP:
    adjnz.w #-1, Cnt_main_stop, CHK_LOOP_MAX ; Counting main clock stop
    jmp    ONCHIP_OSC

CHK_MAIN_OSC:
    mov.w #CNT_CHK_XIN, Cnt_main_stop ; Reset main clock stop counter

CHK_LOOP_MAX:
    adjnz.w #-1, Cnt_tmp, CHK_XIN_FLG
    jmp    HANDLE_EXCEPTION

;=====
; Switched to the On-chip Oscillator Clock
;=====
ONCHIP_OSC:
END_OSC_STOP_DETECT:
    mov.b #00h, prcr          ; Protect Register (write protected)

    mov.w #00h, _Cnt_ms      ; Clear Control Variables
    bclr  ta0s                ; Stop counting
    mov.b #00h, ta0ic        ; Clear Timer's Interrupt Request Flag
    mov.w #TAO_INTERVAL, ta0 ; Reset Timer Register
    bset  ta0s                ; Start Count Timer
    reit

;=====
; Handle Exception
;=====
HANDLE_EXCEPTION:
    ; In this sample, If an exception occurred, processing jumps to ONCHIP_OSC.
    jmp    ONCHIP_OSC

.END

```

7. 参考ドキュメント

ハードウェアマニュアル
M16C/62P グループハードウェアマニュアル Rev.2.30
(最新版をルネサス テクノロジホームページから入手してください。)

テクニカルアップデート/テクニカルニュース
(最新版をルネサス テクノロジホームページから入手してください。)

8. ホームページとサポート窓口

ルネサス テクノロジホームページ
<http://www.renesas.com/jpn/>

M16C ファミリ MCU 技術サポート窓口
E-mail: support_apl@renesas.com

Appendix A. 発振停止、再発振検出機能 関連レジスタ

本サンプルプログラムは、以下のハードウェアマニュアルを参照しています。

・参照ハードウェアマニュアル：M16C/62P グループハードウェアマニュアル Rev.2.30

各レジスタの仕様や機能については、必ず最新のハードウェアマニュアルを参照、確認してください。

A.1. 発振停止検出レジスタ

発振停止検出レジスタ(注1)

| シンボル | アドレス | リセット後の値 | |
|--------------|---|---|----|
| CM2 | 000Ch番地 | 0X000000b(注11) | |
| ビットシンボル | ビット名 | 機能 | RW |
| CM20 | 発振停止、再発振検出許可ビット(注7、9、10、11) | 0：発振停止、再発振検出機能無効 1：発振停止、再発振検出機能有効 | RW |
| CM21 | システムクロック選択ビット2(注2、3、6、8、11、12) | 0：メインクロックまたはPLLクロック 1：オンチップオシレータクロック(オンチップオシレータ発振) | RW |
| CM22 | 発振停止、再発振検出フラグ(注4) | 0：メインクロック停止、再発振を未検出 1：メインクロック停止、再発振を検出 | RW |
| CM23 | XINモニタフラグ(注5) | 0：メインクロック発振 1：メインクロック停止 | RO |
| — (b5-b4) | 予約ビット | “0”にしてください | RW |
| — (b6) | 何も配置されていない。書く場合、“0”を書いてください。読んだ場合、その値は不定。 | | — |
| CM27 | 発振停止、再発振検出時の動作選択ビット(注11) | 0：発振停止検出リセット 1：発振停止、再発振検出割り込み | RW |

注1. このレジスタは、PRCRレジスタのPRC0ビットを“1”(書き込み許可)にした後で書き換えてください。

注2. CM20ビットが“1”(発振停止、再発振検出機能有効)、CM27ビットが“1”(発振停止、再発振検出割り込み)、CPUクロック源がメインクロックのとき、メインクロック停止が検出されるとCM21ビットは“1”(オンチップオシレータクロック)になります。

注3. CM20ビットが“1”で、かつCM23ビットが“1”(メインクロック停止)のとき、CM21ビットを“0”にしないでください。

注4. メインクロック停止検出時とメインクロック再発振検出時“1”になります。このビットが“0”から“1”に変化すると発振停止、再発振検出割り込み要求が発生します。割り込みルーチンで発振停止、再発振検出割り込みと、ウォッチドッグタイマ割り込みの要因判別のために使用してください。プログラムで“0”を書くと“0”になります(“1”を書いても変化しません。また、発振停止、再発振検出割り込み要求が受け付けられても、“0”になりません)。

CM22ビットが“1”のとき、発振停止または再発振を検出しても、発振停止、再発振検出割り込みは発生しません。

注5. 発振停止、再発振検出割り込みルーチンで、CM23ビットを数回読むことによりメインクロックの状態を判定してください。

注6. CM0レジスタのCM07ビットが“0”のとき有効。

注7. PM2レジスタのPM21ビットが“1”(クロック変更禁止)の場合、CM20ビットに書いても変化しません。

注8. CM20ビットが“1”(発振停止、再発振検出機能有効)、CM27ビットが“1”(発振停止、再発振検出割り込み)、CM11ビットが“1”(CPUクロック源はPLLクロック)の場合、メインクロック停止を検出してもCM21ビットは変化しません。この条件でCM22ビットが“0”ならばメインクロック停止検出時に発振停止、再発振検出割り込み要求が発生しますので、割り込みルーチン内でCM21ビットを“1”(オンチップオシレータクロック)にしてください。

注9. ストップモードへ移行する場合、CM20ビットを“0”(無効)にしてください。ストップモードからの復帰後、改めてCM20ビットを“1”(有効)にしてください。

注10. CM0レジスタのCM05ビットを“1”(メインクロック停止)にする前にCM2レジスタのCM20ビットを“0”(無効)にしてください。

注11. CM20、CM21、CM27ビットは発振停止検出リセット時は変化しません。

注12. CM21ビットが“0”(オンチップオシレータ停止)、CM05ビットが“1”(メインクロックを停止)のとき、CM06ビットが“1”(8分周モード)、CM15ビットが“1”(駆動能力HIGH)に固定されます。

A.2. システムクロック制御レジスタ0

システムクロック制御レジスタ0(注1)

| シンボル CM0 | アドレス 0006h番地 | リセット後の値 01001000b | |
|-------------|------------------------------------|---|----|
| ビット シンボル | ビット名 | 機能 | RW |
| CM00 | クロック出力機能選択ビット (シングルチップモード時のみ有効) | b1 b0 0 0 : 入出力ポートP5_7 | RW |
| CM01 | | 0 1 : fCを出力 1 0 : f8を出力 1 1 : f32を出力 | RW |
| CM02 | ウェイトモード時周辺機能 クロック停止ビット(注10) | 0 : ウェイトモード時、周辺機能クロック 停止しない 1 : ウェイトモード時、周辺機能クロック 停止する(注8) | RW |
| CM03 | XCIN-XCOUT駆動能力選択 ビット(注2) | 0 : Low 1 : High | RW |
| CM04 | ポートXC切り替えビット (注2) | 0 : 入出力ポートP8_6、P8_7 1 : XCIN-XCOUT発振機能(注9) | RW |
| CM05 | メインクロック停止ビット (注3、10、12、13) | 0 : 発振 1 : 停止(注4、5) | RW |
| CM06 | メインクロック分周比選択 ビット0(注7、13、14) | 0 : CM16、CM17ビット有効 1 : 8分周モード | RW |
| CM07 | システムクロック選択 ビット(注6、10、11、12) | 0 : メインクロック、PLLクロック またはオンチップオシレータクロック 1 : サブクロック | RW |

注1. このレジスタは、PRCRレジスタのPRC0ビットを“1”(書き込み許可)にした後で書き換えてください。

注2. CM04ビットを“0”(入出力ポート)の間、またはストップモードへ移行したとき、CM03ビットは“1”(HIGH)になります。

注3. このビットは低消費電力モード、またはオンチップオシレータ低消費電力モードにするときに、メインクロックを停止させるためのビットです。メインクロックが停止したかどうかの検出には使用できません。メインクロックを停止させる場合、次のようにしてください。

- (1) サブクロックが安定して発振している状態で、CM07ビットを“1”(サブクロック選択)にする、またはCM2レジスタのCM21ビットを“1”(オンチップオシレータ選択)にする
- (2) CM2レジスタのCM20ビットを“0”(発振停止、再発振検出機能無効)にする
- (3) CM05ビットを“1”(停止)にする

注4. 外部クロック入力時は、0”(発振)にしてください。

注5. CM05ビットが“1”の場合、XOUT端子は“H”になります。また、内蔵している帰還抵抗は接続したままであるので、XIN端子は帰還抵抗を介して、XOUT(“H”)にプルアップされた状態となります。

注6. CM04ビットを“1”(XCIN-XCOUT発振機能)にし、サブクロックの発振が安定した後に、CM07ビットを“0”から“1”(サブクロック)にしてください。

注7. 高速モード、中速モード、オンチップオシレータモード、またはオンチップオシレータ低消費電力モードからストップモードへの移行時、CM06ビットは“1”(8分周モード)になります。

注8. fC32は停止しません。低速モードまたは低消費電力モード時は“1”(ウェイトモード時、周辺機能クロック停止する)にしないでください。

注9. サブクロックを使用する場合、このビットを“1”にしてください。また、ポートP8_6、P8_7は入力ポートで、プルアップなしにしてください。

注10. PM2レジスタのPM21ビットが“1”(クロック変更禁止)の場合、CM02、CM05、CM07ビットに書いても変化しません。

注11. PM21ビットを“1”にする場合、CM07ビットを“0”(メインクロック)にした後で、PM21ビットを“1”にしてください。

注12. CPUクロックのクロック源をメインクロックにする場合、次のようにしてください。

- (1) CM05ビットを“0”(発振)にする。
- (2) メインクロック発振安定時間を待つ。
- (3) CM11ビットを“0”、CM21ビットを“0”、CM07ビットを“0”にする。

注13. CM21ビットが“0”(オンチップオシレータ停止)、CM05ビットが“1”(メインクロックを停止)のとき、CM06ビットが“1”(8分周モード)、CM15ビットが“1”(駆動能力HIGH)に固定されます。

注14. オンチップオシレータモードから高速、中速モードに戻すときは、CM06ビットを“1”、CM15ビットを“1”にしてください。

A.3. システムクロック制御レジスタ 1

システムクロック制御レジスタ1(注1)

| ビット シンボル | ビット名 | 機能 | RW |
|--------------|---------------------------|---|----|
| CM10 | 全クロック停止制御ビット (注4、6) | 0 : クロック発振 1 : 全クロック停止(ストップモード) | RW |
| CM11 | システムクロック選択ビット1 (注6、7) | 0 : メインクロック 1 : PLLクロック(注5) | RW |
| — (b4-b2) | 予約ビット | "0" にしてください。 | RW |
| CM15 | XIN-XOUT駆動能力選択ビット (注2) | 0 : LOW 1 : HIGH | RW |
| CM16 | メインクロック分周比 選択ビット1(注3) | b7 b6 0 0 : 分周なしモード 0 1 : 2分周モード 1 0 : 4分周モード 1 1 : 16分周モード | RW |
| CM17 | | | RW |

- 注1. このレジスタは、PRCRレジスタのPRC0ビットを“1”(書き込み許可)にした後で書き換えてください。
- 注2. 高速モード、中速モードからストップモードへの移行時、または低速モードで、CM05ビットを“1”(メインクロック停止)にしたとき、CM15ビットは“1”(駆動能力HIGH)になります。
- 注3. CM06ビットが“0”(CM16、CM17ビット有効)の場合、有効となります。
- 注4. CM10ビットが“1”(ストップモード)の場合、XOUTは“H”となり、内蔵している帰還抵抗は切り離されません。XCIN端子、XCOUT端子は、ハイインピーダンスになります。CM11ビットが“1”(PLLクロック)、またはCM2レジスタのCM20ビットが“1”(発振停止検出機能有効)の場合、CM10ビットを“1”にしないでください。
- 注5. PLC0レジスタのPLC07ビットを“1”(PLL動作)にした後、tsu(PLL)待って、CM11ビットを“1”(PLLクロック)にしてください。
- 注6. PM2レジスタのPM21ビットが“1”(クロック変更禁止)の場合、CM10、CM11ビットに書いても変化しません。PM2レジスタのPM22ビットが“1”(ウォッチドッグタイマのカウントソースはオンチップオシレータクロック)の場合、CM10ビットに書いても変化しません。
- 注7. CM07=“0”、CM21=“0”のとき有効。

A.4. 発振停止、再発振検出割り込みの動作(CM27ビット=1)

サンプルプログラムでは、CPU クロック源として PLL クロックを使用しています。
この場合の発振停止、再発振検出割り込みの動作を以下に示します。

メインクロックが停止した場合、次の状態になります。

- ・ 発振停止、再発振検出割り込み要求が発生する。
- ・ CM22 ビット=1 (メインクロック停止を検出)
- ・ CM23 ビット=1 (メインクロック停止)
- ・ CM21 ビットは変化しない

(注)PLL クロックを CPU クロック源として使用している場合、

メインクロックが停止しても CM21 ビットは変化しませんので、

割り込みルーチン内で"1" (オンチップオシレータクロック(オンチップオシレータ発振)) にしてください。

メインクロックが停止した状態から再発振した場合、次の状態になります。

- ・ 発振停止、再発振検出割り込み要求が発生する。
- ・ CM22 ビット=1 (メインクロック再発振を検出)
- ・ CM23 ビット=0 (メインクロック発振)
- ・ CM21 ビットは変化しない

A.5. 発振停止、再発振検出機能使用方法

- ・ 発振停止、再発振検出割り込みは、ウォッチドックタイマ割り込みとベクタを共用しています。
発振停止、再発振検出割り込みとウォッチドッグタイマ割り込みの両方を使用する場合、割り込みルーチンで CM22 ビット (発振停止、再発振検出フラグ) を読み、どちらの割り込み要因による割り込み要求かを判定してください。
- ・ 発振停止後、メインクロックが再発振した場合は、プログラムでメインクロックを CPU クロックや周辺機能のクロック源に戻してください。
- ・ 発振停止、再発振検出割り込み発生と同時に CM22 ビットが"1"になります。
CM22 ビットが"1"のとき、発振停止、再発振検出割り込みは禁止されます。
プログラムで CM22 ビットを"0"にすると、発振停止、再発振検出割り込みが許可されます。
- ・ 低速モード時、CM20 ビットが"1"で、メインクロックが停止すると、発振停止、再発振検出割り込み要求が発生します。同時にオンチップオシレータが発振を開始します。このとき、CPU クロックはサブクロックのままですが、周辺機能クロックはオンチップオシレータクロックがクロック源になります。
- ・ 発振停止、再発振検出機能を使用中にウェイトモードへ移行する場合は、CM02 ビットを"0" (ウェイトモード時周辺機能クロックを停止しない) にしてください。
- ・ 発振停止、再発振検出機能は外部要因によるメインクロック停止に備えた機能ですので、プログラムでメインクロックを停止または発振させる場合、すなわちストップモードにする、または CM05 ビットを変更する場合は CM20 ビットを"0" (発振停止、再発振検出機能無効) にしてください。
- ・ メインクロックの周波数が 2MHz 以下の場合、この機能は使用できませんので、CM20 ビットを"0"にしてください。

改訂記録

| Rev. | 発行日 | 改訂内容 | |
|------|------------|-------|--|
| | | ページ | ポイント |
| 1.00 | 2005.02.25 | - | 初版発行 |
| 1.01 | 2005.06.22 | 7 | 内容が重複している説明文を削除 |
| | | 22,23 | Appendix A. 発振停止検出レジスタとシステムクロック制御レジスタ 0 の誤記修正 |
| | | | |

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジー製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジーが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジーは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジーは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジー半導体製品のご購入に当たりましては、事前にルネサス テクノロジー、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジーホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したものです。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジーはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジーは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジー、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジーの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらルネサス テクノロジー、ルネサス販売または特約店までご照会ください。