

お客様各位

カタログ等資料中の旧社名の扱いについて

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

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

2010年4月1日
ルネサスエレクトロニクス株式会社

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

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

ご注意書き

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

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

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

SH7730 グループ

TMU 設定例

要旨

この資料は、タイマユニット機能 (TMU) の設定例を掲載しています。

動作確認デバイス

SH7730

目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. TMU レジスタ設定手順.....	4
4. 参考プログラムについて.....	5
5. 参考プログラム例.....	16
6. 参考ドキュメント.....	30

2. 応用例の説明

本参考プログラムではタイマユニット (TMU チャンネル 0) を使用し、500ms 間隔でアンダフローを発生させ、アンダフロー割り込み処理で LED の出力反転処理を行います。

2.1 使用機能の動作概要

タイマユニット (TMU) は、タイマカウンタ (TCNT) をカウントダウンさせるクロックで、タイマカウンタ (TCNT) に設定されている値をカウントダウンさせ、アンダフロー (H'00000000 → H'FFFFFFF) が発生するまでの時間で時間計測を行います。

TCNT をカウントダウンさせるクロックは、Pφ (周辺クロック) とタイマコントロールレジスタ (TCR) の TPSC[2:0] の設定値により決まります。

アンダフロー (H'00000000 → H'FFFFFFF) が発生すると、対応するチャンネルの TCR のアンダフローフラグ (UNF) がセットされます。このとき、TCR の UNIE ビットが B'1 ならば、CPU に割り込みを要求します。また、同時に TCNT カウンタ自体には、TCOR の値がセットされ、セットされた値からカウントダウン動作を続けます。表 1 に TMU の概要を示します。

TMU 機能の詳細については、「SH7730 グループ ハードウェアマニュアル (RJJ09B0339) 18 章 タイマユニット (TMU)」を参照ください。

表 1 TMU の概要

項目	概要
チャンネル数	3 チャンネル
カウンタサイズ	32 ビットカウンタ
クロックソース	Pφ/4 or Pφ/16 or Pφ/64 or Pφ/256 or Pφ/1024 Pφ: 周辺クロック
アンダフローフラグ	TCNT がアンダフローすると対応するチャンネルの TCR.UNF フラグがセットされます。 UNF フラグがセットされる詳細なタイミングについては、SH7730 グループ ハードウェアマニュアル (RJJ09B0339) 18 章 タイマユニット (TMU) 18.4.1 ステータスフラグのセットタイミング」を参照ください。
割り込み要求	アンダフロー割り込み (TUNI) 制御 TUNI がセットされる詳細なタイミングについては、SH7730 グループ ハードウェアマニュアル (RJJ09B0339) 18 章 タイマユニット (TMU) 18.4.1 ステータスフラグのセットタイミング」を参照ください。

3. TMU レジスタ設定手順

TMU 開始時のレジスタ設定フローについては図 1，TMU 停止時のレジスタ設定フローについては図 2 に記載します。

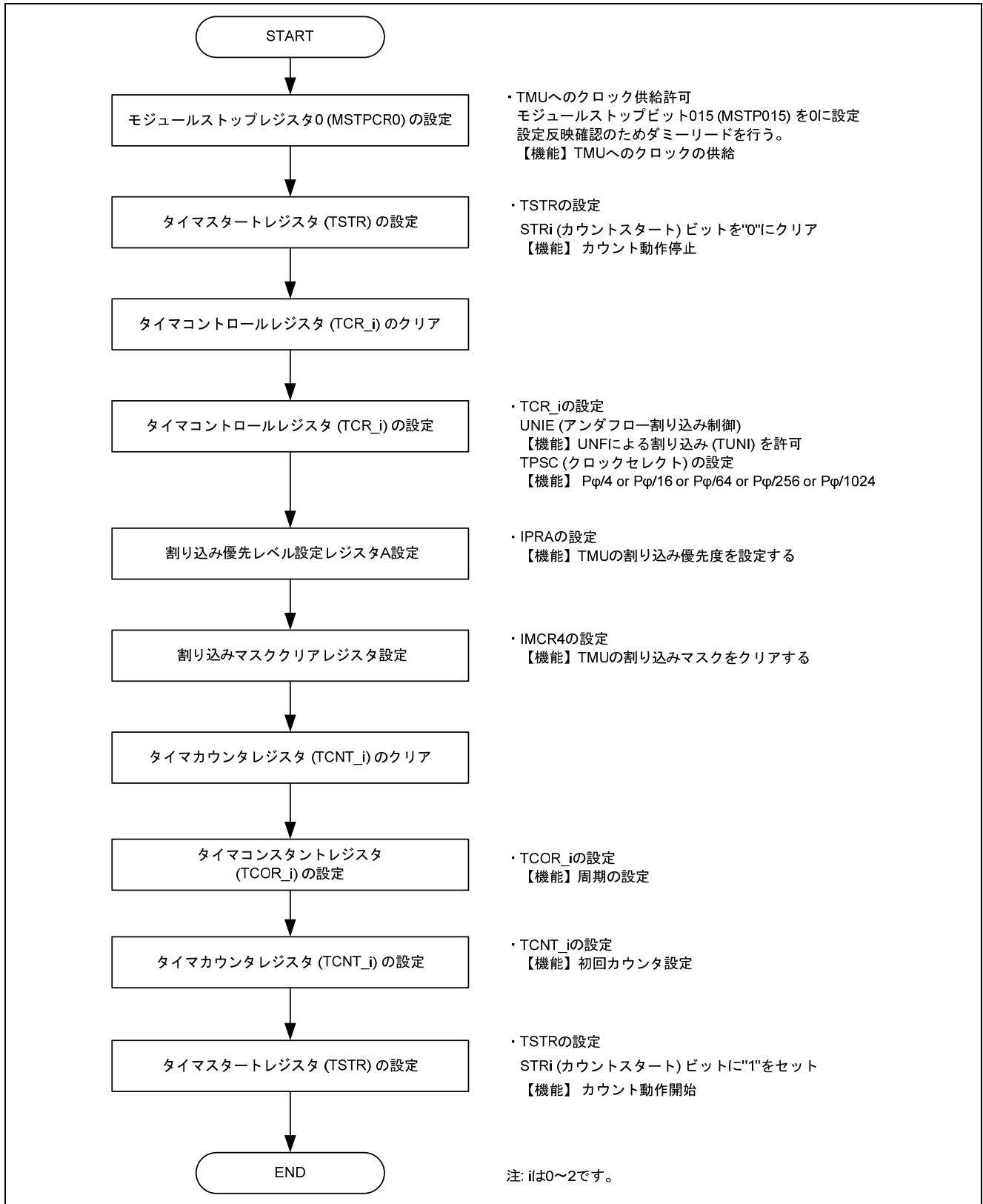


図 1 TMU 開始時レジスタ設定フロー

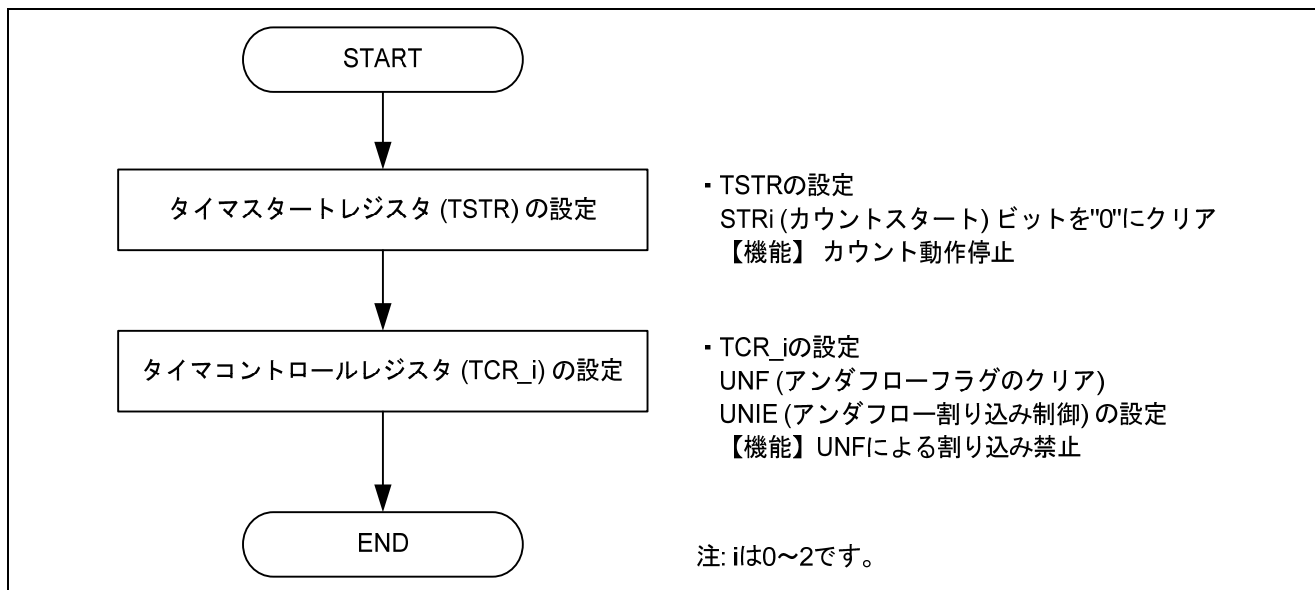


図 2 TMU 停止時レジスタ設定フロー

4. 参考プログラムについて

参考プログラムの動作は、500ms ごとに LED を点滅させ、10 回の LED 点灯後、LED 点滅を停止します。本参考プログラムでは、タイマユニット (TMU0) を 500ms のタイマとして使用し、アンダフロー割り込み (TUNI) で、アンダフローフラグ (TCR.UNF) のクリアと LED の出力反転処理をしています。10 回の LED 点灯後、タイマの停止を行っています。

4.1 参考プログラムの動作仕様

表 2 参考プログラム動作仕様

項目	概要
使用チャンネル	0 チャンネル
カウンタサイズ	32 ビットカウンタ (固定)
クロックソース	Pφ/64
割り込み	UNF による割り込み (TUNI) を許可
TMU0 割り込み優先レベル	1
タイマ設定時間	500ms

4.2 参考プログラムのインタフェース

参考プログラムでは、TMU0 を動作させるために、以下のインタフェースを作成します。

(1) TMU0 タイマ生成処理 (tmu0_create())

【概要】

TMU0 タイマの生成処理を行います。このインタフェースでは、パラメータの設定値により、クロックセレクト (TCR0.TPSC)、割り込み優先レベル (INTC.IPRA) をレジスタに設定します。

【パラメータ】

型	変数	内容
T_TMU_TimerTYPE	mClockSelect	TCR0.TPSC の設定値
	mIntcPri	INTC.IPRA の設定値

(2) TMU0 タイマ開始処理 (tmu0_start())

【概要】

TMU0 タイマを起動します。このインタフェースでは、パラメータのタイマ設定時間より周期を計算し、TCOR に値を設定します。また、アンダフローが発生した際 (タイムアウト時) にコールされる関数の登録を行います。

【パラメータ】

型	変数	内容
T_TMU_SETTCOR_INFO	mTimeValue	タイマ設定時間 (ms) 制限事項 <ul style="list-style-type: none"> ● ms 単位のみ指定可能 ● 1ms ~ 1000ms の範囲指定可能
TMU_TIMEOUT_CALLBACK	void (*TMU_TIMEOUT_CALLBACK) (void)	アンダフロー発生時 (タイムアウト時) のコールバック関数登録

(3) TMU0 タイマ停止処理 (tmu0_stop())

【概要】

TMU0 タイマを停止します。

【パラメータ】

なし。

4.3 参考プログラムの処理フロー

4.3.1 メイン処理フロー

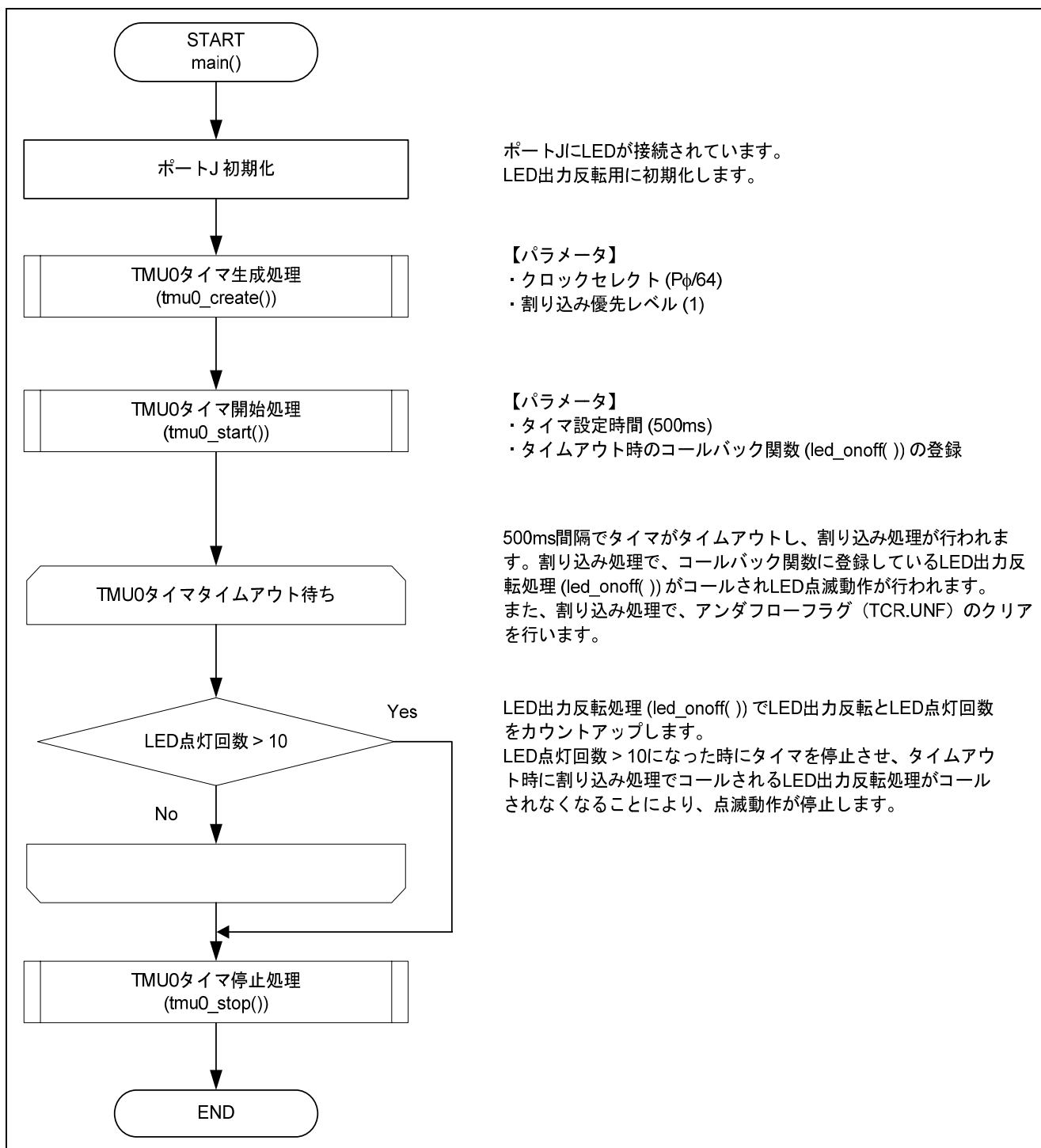


図 3 TMU メイン処理フロー

4.3.2 TMU0 タイマ生成処理

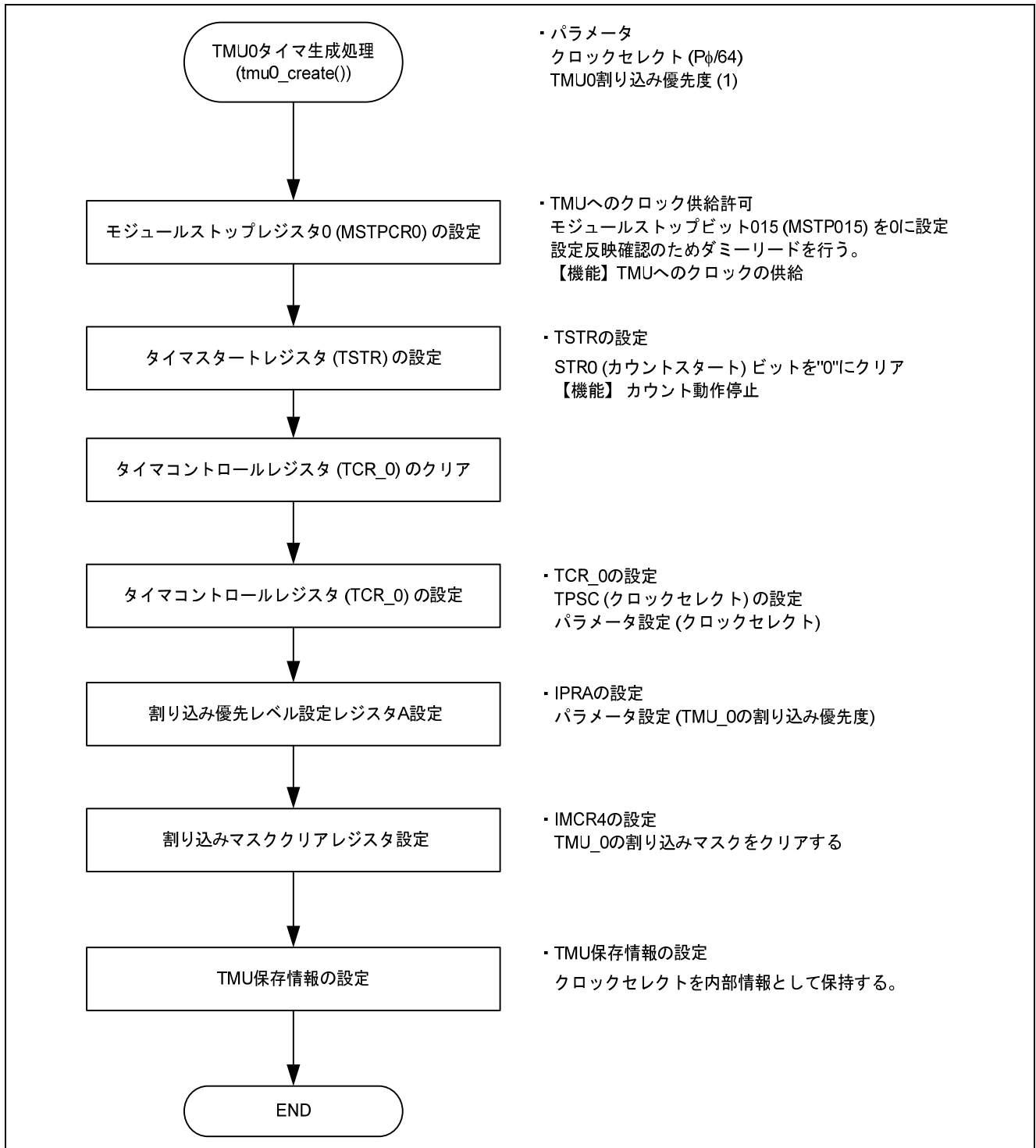


図 4 TMU0 タイマ生成処理フロー

4.3.3 TMU0 タイマ開始処理

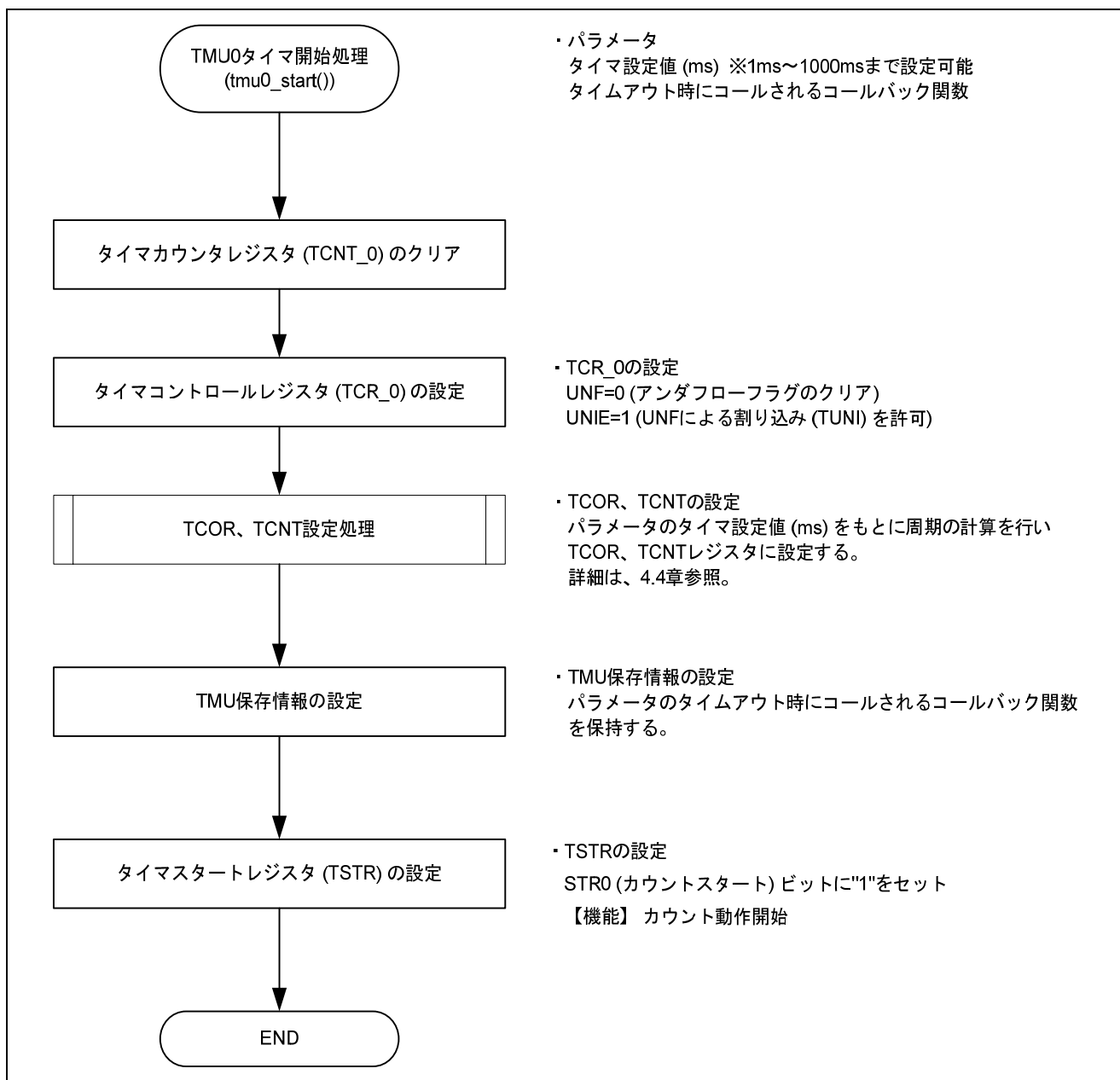


図 5 TMU0 タイマ開始処理フロー

4.3.4 TMU0 タイマ停止処理

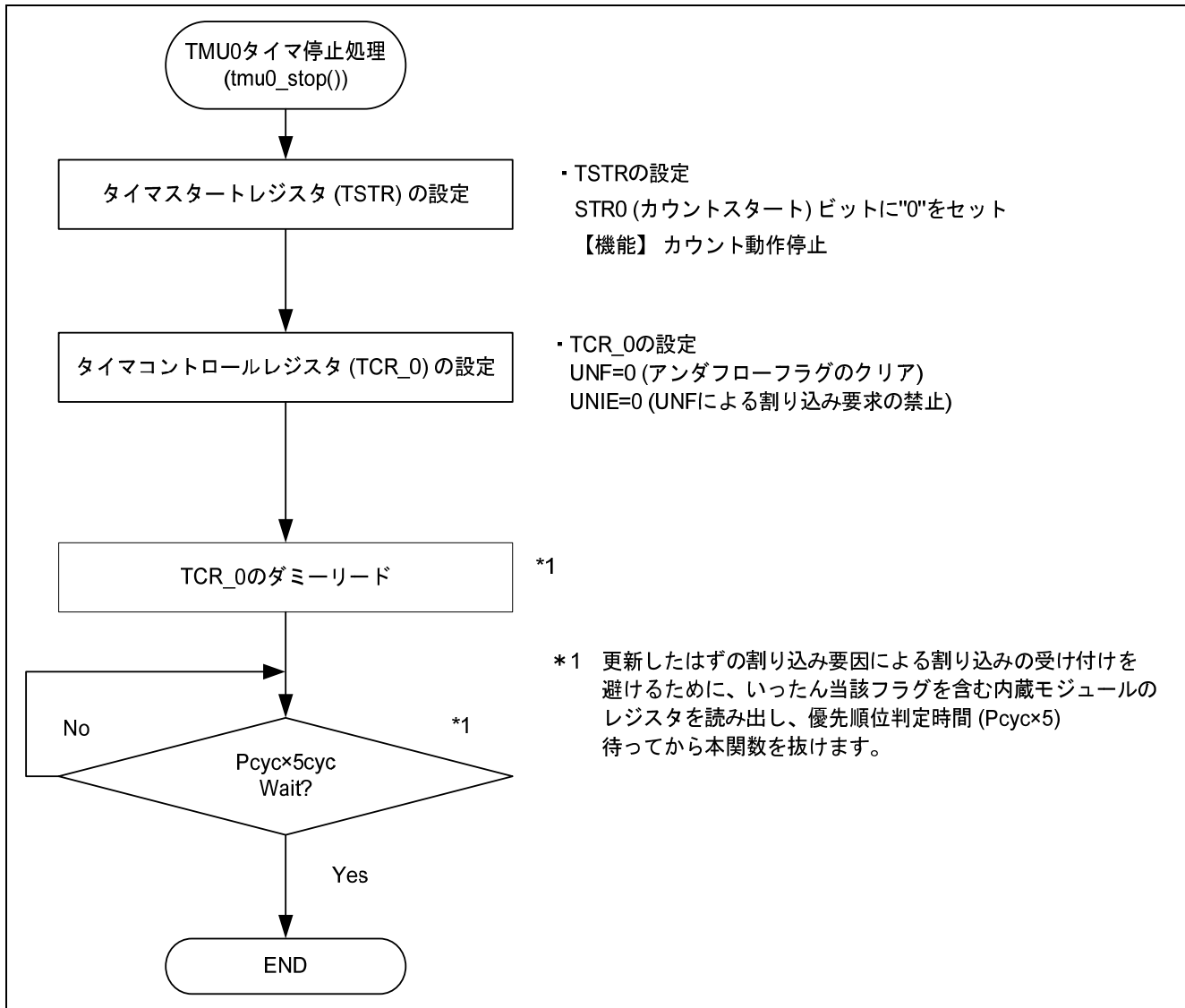


図 6 TMU0 タイマ停止処理フロー

4.3.5 TMU0 割り込み時処理フロー

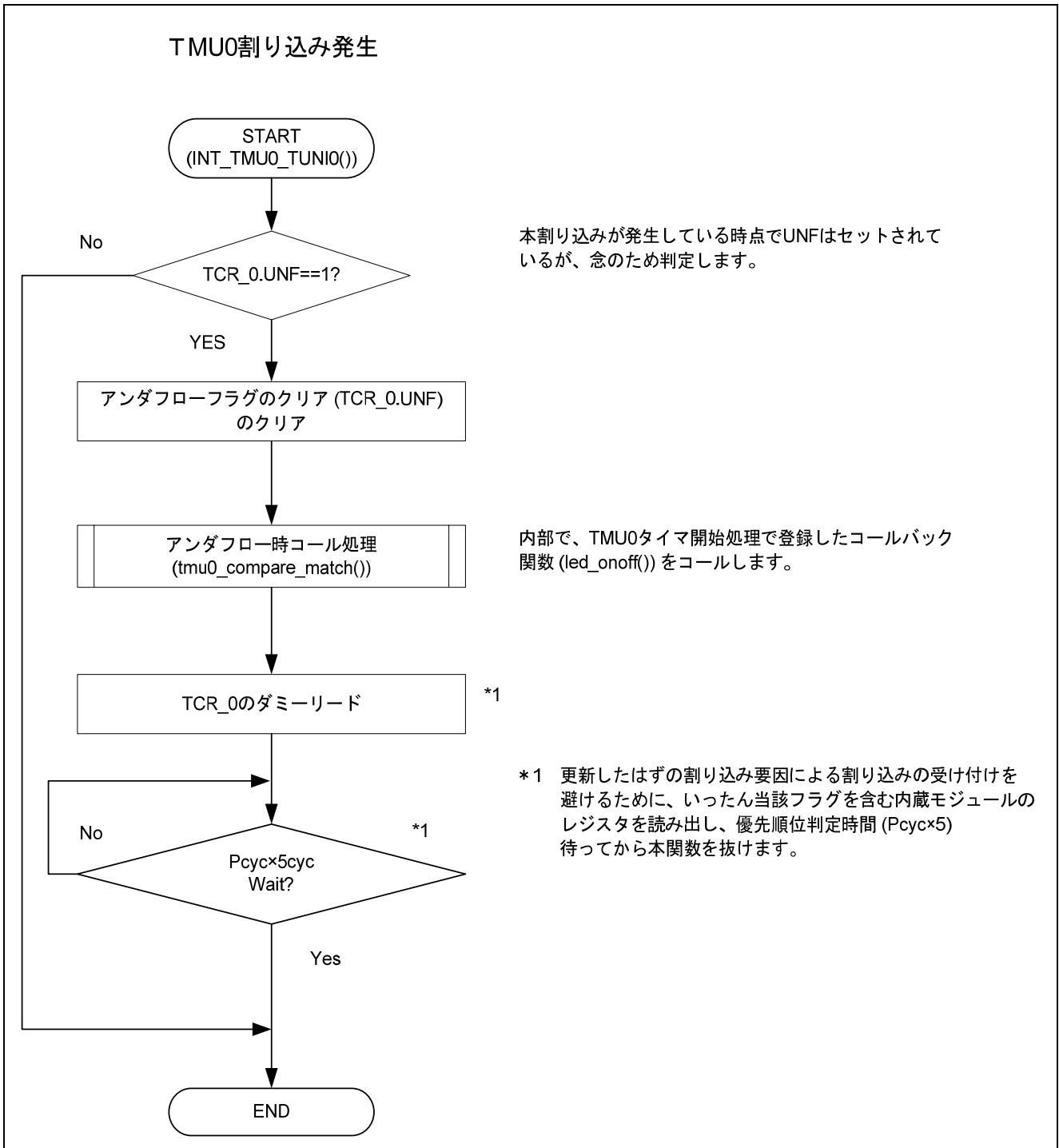


図 7 TMU0 割り込み時処理フロー

4.3.6 コールバック登録関数 (led_onoff ()) の処理フロー

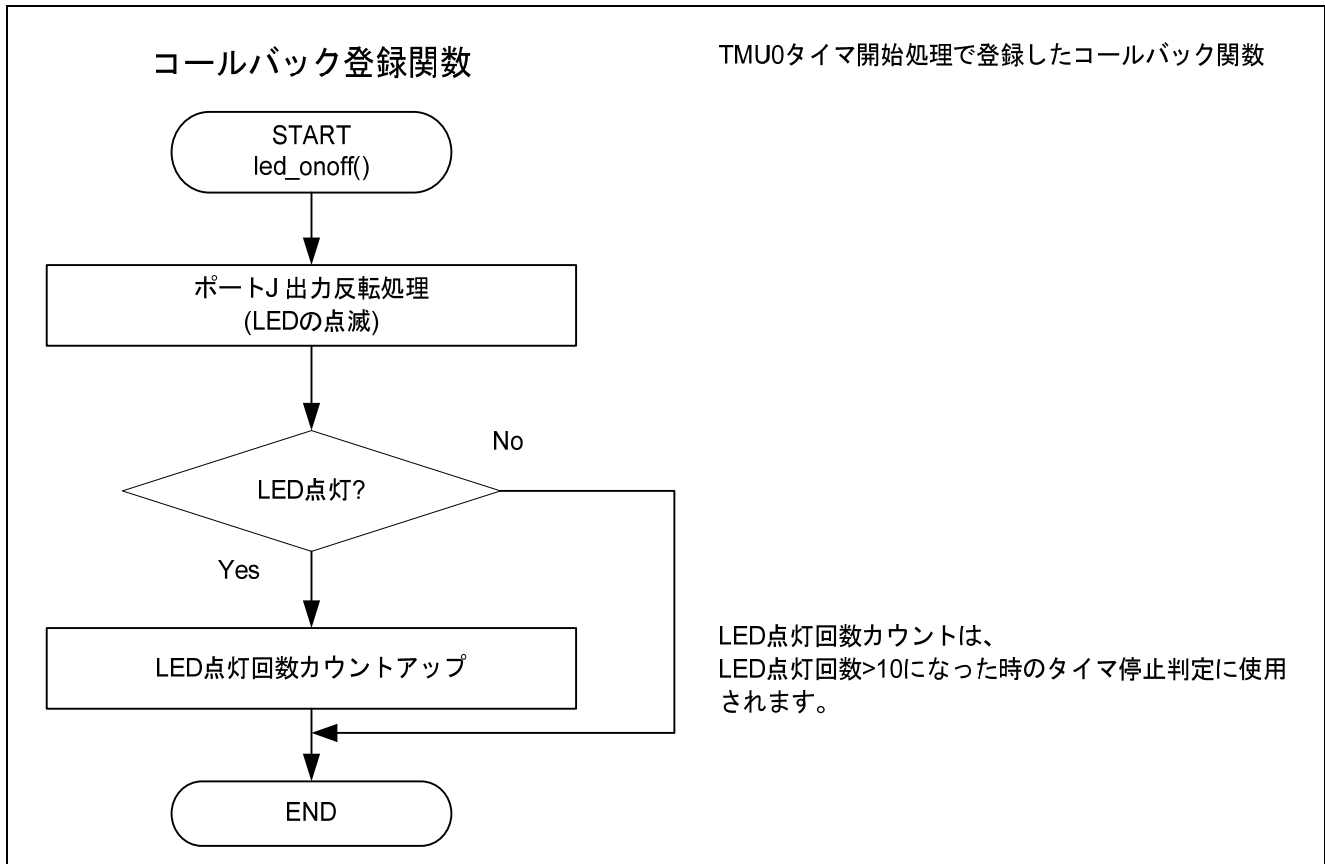


図 8 コールバック登録関数 (led_onoff ())

4.4 参考プログラムのレジスタ設定値

表 3 に参考プログラムのレジスタ設定を示します。

表 3 参考プログラムのレジスタ設定

レジスタ名	アドレス	設定値	機能と設定値
タイマスタートレジスタ (TSTR)	H'FFD8 0004	H'00	<ul style="list-style-type: none"> タイマ生成時 タイマ停止時 STR[2:0] = B'000: タイマカウント停止
		H'01	<ul style="list-style-type: none"> タイマ開始時 STR[2:0] = B'001: タイマカウント開始
タイマコントロールレジスタ_0 (TCR_0)	H'FFD8 0010	H'0002	<ul style="list-style-type: none"> タイマ生成時 TPSC[2:0] = B'010: P ϕ /64
		H'0022	<ul style="list-style-type: none"> タイマ開始時 UNF = B'0: アンダフローフラグのクリア UNIE = B'1: 割り込み要求を許可
		H'0022	<ul style="list-style-type: none"> TUNI 割り込み時 UNF = B'0: アンダフローフラグのクリア
		H'0002	<ul style="list-style-type: none"> タイマ停止時 UNF = B'0: アンダフローフラグのクリア UNIE = B'0: 割り込み要求を禁止
タイマカウンタ_0 (TCNT_0)	H'FFD8 000C	H'0003 F93F	<ul style="list-style-type: none"> タイマ開始時 260415 (H'3F93F) 回: 約 500ms 詳細は、下記【TCOR 設定値】を参照ください
タイマコンスタントレジスタ_0 (TCOR_0)	H'FFD8 0008	H'0003 F93F	<ul style="list-style-type: none"> タイマ開始時 260415 (H'3F93F) 回: 約 500ms 詳細は、下記【TCOR 設定値】を参照ください
割り込み優先レベル設定レジスタ A (IPRA)	H'A408 0000	H'1000	<ul style="list-style-type: none"> タイマ生成時 IPR0n (ビット名) = 1: 優先レベル 1
割り込みマスククリアレジスタ (IMCR4)	H'A408 00D0	H'10	<ul style="list-style-type: none"> タイマ生成時 IMCRn3 (ビット名) = B'1: 割り込みマスククリア

【TCOR 設定値】

P ϕ = 33.33MHz, クロックセレクト = P ϕ /64 として, タイマカウンタ開始後 500ms 後にアンダフローさせる TCOR の設定値は以下のようになります。

P ϕ = 33.33MHz の場合の 1 クロック時間: 約 30 (ns)
 TCNT が 1 カウントダウンする時間: 30 (ns) () \times 64 = 1920 (ns)
 500ms を ns に変換: 1000000 \times 500 = 500000000 (ns)
 500ms でのカウント数: 500000000 () \div 1920 () \approx 260416 (H'3F940) 回

実際の周期は, 「SH7730 グループ ハードウェアマニュアル 18 章 タイマユニット (TMU) の図 UNF のセットタイミング」より, TCOR に設定した値+1 となります。

そのため, 上記 で求めた値から-1 した値を TCOR に設定します。

$$\text{TCOR 設定値} = 260416 - 1 = 260415 \text{ (H'3F93F)}$$

【周期時間設定 参考例】

$P\phi = 33.33\text{MHz}$, クロックセレクト = $P\phi/64$ として , 周期最大値を計算すると以下のようになります。

TCOR 設定最大値: 4294967295 (H'FFFFFFF)

TCNT が 1 カウントダウンする時間: 上記 より , 1920ns

周期最大時間: $4294967295 \times 1920 \text{ (ns)} = 8246337206400 \text{ (ns)} \approx 8246 \text{ (秒)}$

つまり $P\phi = 33.33 \text{ MHz}$, クロックセレクト = $P\phi/64$ として TCOR に設定できるタイマの最大値は 約 8246 (秒) となります。それ以上のタイマを設定する必要がある場合は , TCR レジスタの TPSC[2:0]ビットで TCNT のカウントクロックを変更するか , 一定期間のタイマを複数回実施する等の対応を行ってください。

4.5 参考プログラムのマクロ定義

表 4 に参考プログラムのマクロ定義を示します。

表 4 参考プログラムで使用しているマクロ定義

マクロ定義	設定値	機能
D_TMU_P_FREQ	H'001E	P ϕ の 1 クロック時間 (ns) P ϕ = 33.33MHz の場合の値を設定しています。 P ϕ が異なる場合は、この設定値を変更してください。
D_TMU_PRI_0 ~ 15	H'00 ~ H'0F	割り込み優先レベル 0 ~ 15 0 指定時はマスクされます。
D_TMU_DIV_4 (P ϕ /8)	H'0000	P ϕ /8
D_TMU_DIV_16 (P ϕ /16)	H'0001	P ϕ /16
D_TMU_DIV_64 (P ϕ /64)	H'0002	P ϕ /64
D_TMU_DIV_256 (P ϕ /256)	H'0003	P ϕ /256
D_TMU_DIV_1024 (P ϕ /1024)	H'0004	P ϕ /1024

5. 参考プログラム例

(1) サンプルプログラムリスト"sh7730.c"

```

1      /*****
2      * DISCLAIMER
3
4      * This software is supplied by Renesas Technology Corp. and is only
5      * intended for use with Renesas products. No other uses are authorized.
6
7      * This software is owned by Renesas Technology Corp. and is protected under
8      * all applicable laws, including copyright laws.
9
10     * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11     * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12     * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13     * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14     * DISCLAIMED.
15
16     * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17     * TECHNOLOGY CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18     * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19     * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20     * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21
22     * Renesas reserves the right, without notice, to make changes to this
23     * software and to discontinue the availability of this software.
24     * By using this software, you agree to the additional terms and
25     * conditions found by accessing the following link:
26     * http://www.renesas.com/disclaimer
27     *****/
28     /* Copyright (C) 2009. Renesas Technology Corp., All Rights Reserved. */
29     /*"FILE COMMENT"***** Technical reference data ***** */
30     * System Name : SH7730 Sample Program
31     * File Name : sh7730.c
32     * Abstract : SH7730 TMU 設定例 Sample Program
33     * Version : Ver 1.00
34     * Device : SH7730
35     * Tool-Chain : High-performance Embedded Workshop (Version 4.04.01.001)
36     * : C/C++ Compiler Package for SuperH Family (V.9.02release00)
37     * OS : None
38     * H/W Platform : アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
39     * Description : SH7730 TMU 設定例のサンプルプログラムです。
40     * :
41     * Operation :
42     * Limitation :
43     * :
44     *****/
45     * History : 03.JUNE.2009 Ver. 1.00 First Release
46     /*"FILE COMMENT END"***** */
47
48     #include <machine.h>
49     #include "iodefine.h"
50     #include "tmu.h"
51
52     /* プロトタイプ宣言 */
53     void main(void);
54     void led_onoff(void);
55
56     /* LED 出力カウンタ */
57     unsigned long g_LedOutCount = 0;
58
59     /*"FUNC COMMENT"*****
60     * ID :
    
```

```

61  * Outline           : TMU タイマサンプル処理
62  * Include           :
63  * Declaration       : void main(void)
64  * Description       : TMU タイマの使用例を示します。
65  *                   : 500ms 間隔でタイマをタイムアウトさせ
66  *                   : LED 出力反転処理を行います。
67  *                   : 10 回の点灯後、タイマを停止します。
68  *                   :
69  * Argument          : none
70  * Return Value      : none
71  * Calling Functions :
72  *""FUNC COMMENT""*****/
73  void main(void)
74  {
75      T_TMU_TimerTYPE      TimerType;
76      T_TMU_SETTCOR_INFO   SetTCORInfo;
77
78      g_LedOutCount = 0;
79      memset(&TimerType, 0x00, sizeof(TimerType));
80      memset(&SetTCORInfo, 0x00, sizeof(SetTCORInfo));
81
82      /* ポート J 初期化 (LED 出力用) */
83      PFC.PJCR.WORD = (PFC.PJCR.WORD & 0xCFFF) | 0x1000; /* PJ6 出力設定 */
84
85      /* タイマ生成用構造体の設定 */
86      TimerType.mClockSelect = D_TMU_DIV_64; /* P /64 */
87      TimerType.mIntcPri = D_TMU_PRI_1; /* 割り込みレベル 1 */
88
89      /* TMU0 タイマ生成処理 */
90      tmu0_create(&TimerType);
91
92      /* タイマ時間設定 */
93      SetTCORInfo.mTimeValue = 500; /* タイマ設定値(500ms) */
94
95      /* TMU0 タイマ開始処理 */
96      tmu0_start(&SetTCORInfo, led_onoff);
97
98      /* led 点灯まで待つ */
99      while(1)
100     {
101         /* led が 10 回点灯したらループを抜ける */
102         if(g_LedOutCount == 10)
103         {
104             /* TMU0 タイマ停止処理 */
105             tmu0_stop();
106             break;
107         }
108     }
109
110     while(1)
111     {
112     }
113
114 }
115
116
117 /*""FUNC COMMENT""*****
118 * ID           :
119 * Outline      : Led 出力反転処理
120 * Include      :
121 * Declaration  : void led_onoff(void)
122 * Description  : Led 出力反転処理をします。
123 *             :
124 * Argument     : none
125 * Return Value : none
126 * Calling Functions :
    
```

```

127  *""FUNC COMMENT END""*****
128  void led_onoff(void)
129  {
130
131      /* ポートJ 出力反転処理 */
132      PORT.PJDR.BYTE ^= 0x40;          /* LED1 点滅 (ポートJ: H   L ) */
133
134      /* 出力回数のカウント */
135      if( PORT.PJDR.BIT.B6 == 1 )
136      {
137          g_LedOutCount++;
138      }
139
140  }
141
142  /* End of File */

```

(2) サンプルプログラムリスト"tmu.c"

```

1      /*****
2      * DISCLAIMER
3
4      * This software is supplied by Renesas Technology Corp. and is only
5      * intended for use with Renesas products. No other uses are authorized.
6
7      * This software is owned by Renesas Technology Corp. and is protected under
8      * all applicable laws, including copyright laws.
9
10     * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11     * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12     * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13     * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14     * DISCLAIMED.
15
16     * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17     * TECHNOLOGY CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18     * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19     * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20     * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21
22     * Renesas reserves the right, without notice, to make changes to this
23     * software and to discontinue the availability of this software.
24     * By using this software, you agree to the additional terms and
25     * conditions found by accessing the following link:
26     * http://www.renesas.com/disclaimer
27     *****/
28     /* Copyright (C) 2009. Renesas Technology Corp., All Rights Reserved. */
29     /*"FILE COMMENT"***** Technical reference data ******/
30     * System Name : SH7730 Sample Program
31     * File Name   : tmu.c
32     * Abstract    : SH7730 TMU 設定例 Sample Program
33     * Version     : Ver 1.00
34     * Device      : SH7730
35     * Tool-Chain  : High-performance Embedded Workshop (Version 4.04.01.001)
36     *             : C/C++ Compiler Package for SuperH Family (V.9.02release00)
37     * OS          : None
38     * H/W Platform : アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
39     * Description : SH7730 TMU 設定例のサンプルプログラムです。
40     *             :
41     * Operation   :
42     * Limitation  :
43     *             :
44     *****/
45     * History     : 03.JUNE.2009 Ver. 1.00 First Release
46     *"FILE COMMENT END"******/
47
48     #include <machine.h>
49     #include "iodefine.h"
50     #include "tmu.h"
51
52     /* TMU 保存情報 */
53     static T_TMU_SAVE_INFO      g_TMU_SAVE_INFO;
54
55     /* クロックセレクト対応テーブル */
56     static const T_TMU_CLOCK_SELECT  gc_Clock_Match_tbl[] =
57     {
58         {D_TMU_DIV_4, D_TMU_DIVValue_4},
59         {D_TMU_DIV_16, D_TMU_DIVValue_16},
60         {D_TMU_DIV_64, D_TMU_DIVValue_64},
61         {D_TMU_DIV_256, D_TMU_DIVValue_256},
62         {D_TMU_DIV_1024, D_TMU_DIVValue_1024}
63     };
    
```

```

64
65  /* ローカル関数宣言 */
66  static void tmu_set_TCOR_TCNT_Type_ms(T_TMU_SETTCOR_INFO *i_pTCorInfo);
67  static T_TMU_PClockDiv tmu_get_ClockDivValue(void);
68  static void tmu0_set_g_TMU_SAVE_INFO(T_TMU_SAVE_INFO *i_pTMU_SAVE_INFO);
69  static void tmu0_get_g_TMU_SAVE_INFO(T_TMU_SAVE_INFO *o_pTMU_SAVE_INFO);
70
71  /*"FUNC COMMENT"*****
72  * ID :
73  * Outline : TMU0 タイマ生成処理
74  * Include :
75  * Declaration : void tmu0_create
76  * : (T_TMU_TimerTYPE *i_pTimerType)
77  * Description : TMU0 の初期設定をする。
78  * :
79  * Argument : T_TMU_TimerTYPE *i_pTimerType :
80  * : タイマ生成用構造体アドレス
81  * Return Value : none
82  * Calling Functions :
83  *"FUNC COMMENT END"*****/
84  void tmu0_create(T_TMU_TimerTYPE *i_pTimerType)
85  {
86      T_TMU_SAVE_INFO ntmu_save_info;
87      unsigned long dummy;
88      unsigned short work;
89
90      memset(&ntmu_save_info, 0x00, sizeof(ntmu_save_info));
91
92      /* ==== モジュールストップレジスタの設定 ==== */
93      LOWP.MSTPCRO &= ~0x00008000;
94
95      dummy = LOWP.MSTPCRO; /* 設定反映確認のためダミリード */
96
97      /* ==== タイマスタートレジスタ (TSTR) の設定 ==== */
98      TMU.TSTR.BIT.STRO = 0; /* カウント停止 */
99
100     /* ==== タイマコントロールレジスタ (TCRO) の設定 ==== */
101     TMU0.TCR = 0x0000; /* 初期化 */
102
103     /* ==== TCR の設定値 ==== */
104     TMU0.TCR |= i_pTimerType->mClockSelect; /* クロックセレクト設定 */
105
106     /* TMU 保存情報にクロックセレクト設定 */
107     ntmu_save_info.mClockSelect = i_pTimerType->mClockSelect;
108
109     /* ==== 割り込み関連設定 ==== */
110     /* 割り込み優先レベル設定レジスタ A 設定(TMU0 有効) */
111     work = INTC0.IPRA & 0x0fff;
112     work |= (i_pTimerType->mIntcPri << 12);
113     INTC0.IPRA = work;
114
115     /* 割り込みマスククリアレジスタ */
116     INTC0.IMCR4 = 0x10;
117
118     /* TMU 保存情報に設定 */
119     tmu0_set_g_TMU_SAVE_INFO(&ntmu_save_info);
120
121 }
122
123 /*"FUNC COMMENT"*****
124 * ID :
125 * Outline : TMU0 タイマ開始処理
126 * Include :
127 * Declaration : void tmu0_start
128 * : (T_TMU_SETTCOR_INFO *i_pTCORInfo,
129 * : TMU_TIMEOUT_CALLBACK i_func)
    
```

```

130 * Description      : TMU0 タイマを開始する。
131 *                  :
132 * Limitation      : 以下の条件を前提としていますので、
133 *                  : 条件に合わない場合は修正してください。
134 *                  : (1)本関数は引数(T_TMU_SETTCOR_INFO)が
135 *                  : ms の値のみ対応しています。
136 *                  : (2)本関数はP が 33.33MHz として対応しています。
137 *                  : 変更する場合は、#define D_TMU_P_FREQ
138 *                  : 変更してください。
139 *                  : (3)タイマ設定値は 1ms ~ 1000ms のみ対応して
140 *                  : います。
141 *                  : 範囲外の場合は、1ms として扱います。
142 *                  :
143 * Argument         : T_TMU_SETTCOR_INFO *i_pTCORInfo:
144 *                  : TCOR 設定情報構造体アドレス
145 *                  : TMU_TIMEOUT_CALLBACK i_func:
146 *                  : タイムアウト時コールバック関数
147 * Return Value     : none
148 * Calling Functions :
149 *""FUNC COMMENT END""******/
150 void tmu0_start(T_TMU_SETTCOR_INFO *i_pTCORInfo, TMU_TIMEOUT_CALLBACK i_func)
151 {
152     T_TMU_SAVE_INFO    ntmu_save_info;
153
154     memset(&ntmu_save_info, 0x00, sizeof(ntmu_save_info));
155
156     /* TMU 保存情報に取得 */
157     tmu0_get_g_TMU_SAVE_INFO(&ntmu_save_info);
158
159     /* ==== タイマカウンタ (TCNT) の設定 ==== */
160     TMU0.TCNT = 0x0000;          /* タイマカウンタクリア*/
161
162     /* ==== タイマコントロールレジスタ (TCR) の設定 ==== */
163     TMU0.TCR &= ~0x0100;        /* アンダーフローフラグのクリア */
164     TMU0.TCR |= 0x0020;         /* UNF による割り込み (TUNI) を許可 */
165
166     /* タイムアウト時コールバック関数登録 */
167     ntmu_save_info.mfunc = i_func;
168
169     /* TCOR、TCNT 設定処理 */
170     tmu_set_TCOR_TCNT_Type_ms(i_pTCORInfo);
171
172     /* TMU 保存情報に設定 */
173     tmu0_set_g_TMU_SAVE_INFO(&ntmu_save_info);
174
175     /* ==== タイマスタートレジスタ (TSTR) の設定 ==== */
176     TMU.TSTR.BIT.STRO = 1;     /* カウント開始 */
177
178 }
179
180 /*""FUNC COMMENT""******/
181 * ID                  :
182 * Outline             : TMU0 タイマ停止処理
183 * Include             :
184 * Declaration         : void tmu0_stop(void)
185 * Description         : TMU0 タイマを停止する。
186 *                     :
187 * Argument            : none
188 * Return Value        : none
189 * Calling Functions   :
190 *""FUNC COMMENT END""******/
191 void tmu0_stop(void)
192 {
193     /* ==== タイマスタートレジスタ (TSTR) の設定 ==== */
194     TMU.TSTR.BIT.STRO = 0;     /* カウント停止 */
195

```

```

196     TMU0.TCR &= ~0x0020;          /* UNF による割り込み (TUNI) を禁止 */
197     TMU0.TCR &= ~0x0100;          /* アンダフローフラグのクリア */
198
199     /* 更新したはずの割り込み要因による割り込みの受け付けを避ける対応 */
200     int_responstime_wait_for_tmu0();
201
202 }
203
204 /*"FUNC COMMENT"*****
205 * ID          :
206 * Outline     : アンダフロー時コール処理
207 * Include     :
208 * Declaration : void tmu0_compare_match(void)
209 * Description : TMU0 タイマを削除する。
210 *             : TCNT がアンダフローを起こした際にコールされる。
211 *             : intrpg.c ファイルの INT_TMU0_TUNIO() よりコールされる。
212 *             :
213 * Argument    : none
214 * Return Value: none
215 * Calling Functions :
216 /*"FUNC COMMENT END"*****/
217 void tmu0_compare_match(void)
218 {
219     T_TMU_SAVE_INFO    ntmu_save_info;
220
221     memset(&ntmu_save_info, 0x00, sizeof(ntmu_save_info));
222
223     /* TMU 保存情報の取得 */
224     tmu0_get_g_TMU_SAVE_INFO(&ntmu_save_info);
225
226     /* タイムアウト時コールバック関数のコール */
227     if(ntmu_save_info.mfunc != NULL)
228     {
229         ntmu_save_info.mfunc();
230     }
231
232 }
233
234 /*"FUNC COMMENT"*****
235 * ID          :
236 * Outline     : TCOR、TCNT 設定処理 (ms 版)
237 * Include     :
238 * Declaration : void tmu_set_TCOR_TCNT_Type_ms
239 *             : (T_TMU_SETTCOR_INFO *i_pTCORInfo)
240 *             :
241 * Description : 引数(ms)情報より TCOR 設定値を計算し TCOR に
242 *             : 設定する。
243 *             :
244 * Limitation  : 以下の条件を前提としていますので、
245 *             : 条件に合わない場合は修正してください。
246 *             : (1)本関数は引数(T_TMU_SETTCOR_INFO)が
247 *             :     ms の値のみ対応しています。
248 *             : (2)本関数は P が 33.33MHz として対応しています。
249 *             :     変更する場合は、#define D_TMU_P_FREQ
250 *             :     変更してください。
251 *             : (3)タイマ設定値は 1ms ~ 1000ms のみ対応して
252 *             :     います。
253 *             :     範囲外の場合は、1ms として扱います。
254 *             :
255 * Argument    : T_TMU_SETTCOR_INFO *i_pTCORInfo:
256 *             :     TCOR 設定情報構造体アドレス
257 * Return Value: none
258 * Calling Functions :
259 /*"FUNC COMMENT END"*****/
260 void tmu_set_TCOR_TCNT_Type_ms(T_TMU_SETTCOR_INFO *i_pTCORInfo)
261 {
    
```



```

262     unsigned long   nTCOR = 0;
263     unsigned long   time1clock = 0;
264
265     /* タイマ設定値が条件を満たさない場合は、1 とする */
266     if(i_pTCORInfo->mTimeValue <= 0 || i_pTCORInfo->mTimeValue > 1000)
267     {
268         i_pTCORInfo->mTimeValue = 1;
269     }
270
271     /* ms 単位入力値を ns 単位に変更 */
272     nTCOR = i_pTCORInfo->mTimeValue * 1000000;
273
274     /* P /n 時の 1 クロック時間(ns 単位)を計算 */
275     /* P 1 クロック時間(33.33MHz の場合 : 約 30ns) × P 分割値 */
276     time1clock = D_TMU_P_FREQ * tmu_get_ClockDivValue();
277
278     /* TCOR 設定値計算 */
279     nTCOR = nTCOR / time1clock;
280
281     /* 周期は設定した値 + 1 となるため -1 する */
282     if(nTCOR > 0)
283     {
284         nTCOR--;
285     }
286
287     /* ==== タイマコンスタントレジスタ (TCOR) の設定 ==== */
288     TMUO.TCOR = nTCOR;
289
290     /* ==== タイマカウンタ (TCNT) の設定 ==== */
291     TMUO.TCNT = nTCOR;          /* 初期値を設定 */
292
293 }
294
295 /*"FUNC COMMENT"*****
296 * ID                :
297 * Outline           : P 分割値取得処理
298 * Include           :
299 * Declaration       : T_TMU_PClockDiv tmu_get_ClockDivValue(void)
300 * Description       : クロックセレクト対応テーブルより、
301 *                   : P 分割値を取得する。
302 *                   :
303 * Argument          : none
304 * Return Value      : T_TMU_PClockDiv : P 分割値
305 * Calling Functions :
306 *"FUNC COMMENT END"*****/
307 T_TMU_PClockDiv tmu_get_ClockDivValue(void)
308 {
309     T_TMU_SAVE_INFO    ntmu_save_info;
310     int                 i;
311     int                 nsize;
312
313     tmuo_get_g_TMU_SAVE_INFO(&ntmu_save_info);
314
315     nsize = sizeof(gc_Clock_Match_tbl) / sizeof(T_TMU_CLOCK_SELECT);
316
317     for(i = 0; i < nsize; i++)
318     {
319         if(gc_Clock_Match_tbl[i].mClockSetting == ntmu_save_info.mClockSelect)
320         {
321             return gc_Clock_Match_tbl[i].mClockDivValue;
322         }
323     }
324 }
325
326 /* エラーケース (ありえないケース)*/
327 ntmu_save_info.mClockSelect = D_TMU_DIV_4;
    
```

```

328
329     /* TMU 保存情報に設定 */
330     tmu0_set_g_TMU_SAVE_INFO(&ntmu_save_info);
331
332     return D_TMU_DIVValue_4;
333 }
334
335 /*"FUNC COMMENT"*****
336 * ID :
337 * Outline : TMU 保存情報設定処理
338 * Include :
339 * Declaration : void tmu0_set_g_TMU_SAVE_INFO
340 * : (T_TMU_SAVE_INFO *i_pTMU_SAVE_INFO)
341 * Description : TMU 保存情報を設定する。
342 * :
343 * Argument : T_TMU_SAVE_INFO *i_pTMU_SAVE_INFO
344 * : : TMU 保存情報
345 * Return Value : none
346 * Calling Functions :
347 *"FUNC COMMENT END"*****/
348 void tmu0_set_g_TMU_SAVE_INFO(T_TMU_SAVE_INFO *i_pTMU_SAVE_INFO)
349 {
350     memcpy(&g_TMU_SAVE_INFO, i_pTMU_SAVE_INFO, sizeof(T_TMU_SAVE_INFO));
351 }
352
353 /*"FUNC COMMENT"*****
354 * ID :
355 * Outline : TMU 保存情報取得処理
356 * Include :
357 * Declaration : void tmu0_get_g_TMU_SAVE_INFO
358 * : (T_TMU_SAVE_INFO *o_pTMU_SAVE_INFO)
359 * Description : TMU 保存情報を取得する。
360 * :
361 * Argument : T_TMU_SAVE_INFO *o_pTMU_SAVE_INFO
362 * : : TMU 保存情報
363 * Return Value : none
364 * Calling Functions :
365 *"FUNC COMMENT END"*****/
366 void tmu0_get_g_TMU_SAVE_INFO(T_TMU_SAVE_INFO *o_pTMU_SAVE_INFO)
367 {
368     memcpy(o_pTMU_SAVE_INFO, &g_TMU_SAVE_INFO, sizeof(g_TMU_SAVE_INFO));
369 }
370
371 /* End of File */
    
```

(3) サンプルプログラムリスト"tmu.h"

```

1      /******
2      * DISCLAIMER
3
4      * This software is supplied by Renesas Technology Corp. and is only
5      * intended for use with Renesas products. No other uses are authorized.
6
7      * This software is owned by Renesas Technology Corp. and is protected under
8      * all applicable laws, including copyright laws.
9
10     * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11     * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12     * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13     * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14     * DISCLAIMED.
15
16     * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17     * TECHNOLOGY CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18     * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19     * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20     * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21
22     * Renesas reserves the right, without notice, to make changes to this
23     * software and to discontinue the availability of this software.
24     * By using this software, you agree to the additional terms and
25     * conditions found by accessing the following link:
26     * http://www.renesas.com/disclaimer
27     *****/
28     /* Copyright (C) 2009. Renesas Technology Corp., All Rights Reserved. */
29     /*"FILE COMMENT"***** Technical reference data ******/
30     * System Name : SH7730 Sample Program
31     * File Name   : tmu.h
32     * Abstract    : SH7730 TMU 設定例 Sample Program
33     * Version     : Ver 1.00
34     * Device      : SH7730
35     * Tool-Chain  : High-performance Embedded Workshop (Version 4.04.01.001)
36     *             : C/C++ Compiler Package for SuperH Family (V.9.02release00)
37     * OS          : None
38     * H/W Platform : アルファプロジェクト製 SH-4A ボード 型番 AP-SH4A-1A
39     * Description : SH7730 TMU 設定例のサンプルプログラムです。
40     *             :
41     * Operation   :
42     * Limitation  :
43     *             :
44     *****/
45     * History     : 03.JUNE.2009 Ver. 1.00 First Release
46     *"FILE COMMENT END"******/
47
48     #include <stdio.h>
49
50     /* ==== マクロ定義 ==== */
51     #define D_TMU_P_FREQ30          /* P 1クロック時間(33.33MHzの場合:30ns) */
52
53     /* クロックセレクト */
54     typedef enum {
55
56         D_TMU_DIV_4 = 0,          /* P /4 */
57         D_TMU_DIV_16,           /* P /16 */
58         D_TMU_DIV_64,          /* P /64 */
59         D_TMU_DIV_256,         /* P /256 */
60         D_TMU_DIV_1024         /* P /1024 */
61
62     }T_TMU_ClockSelect;
63
    
```

```

64  /* TMU_割り込み優先レベル */
65  typedef enum {
66
67      D_TMUIPRI_0 = 0,          /* 優先レベル0(割り込みマスク) */
68      D_TMUIPRI_1,            /* 優先レベル1 */
69      D_TMUIPRI_2,            /* 優先レベル2 */
70      D_TMUIPRI_3,            /* 優先レベル3 */
71      D_TMUIPRI_4,            /* 優先レベル4 */
72      D_TMUIPRI_5,            /* 優先レベル5 */
73      D_TMUIPRI_6,            /* 優先レベル6 */
74      D_TMUIPRI_7,            /* 優先レベル7 */
75      D_TMUIPRI_8,            /* 優先レベル8 */
76      D_TMUIPRI_9,            /* 優先レベル9 */
77      D_TMUIPRI_10,           /* 優先レベル10 */
78      D_TMUIPRI_11,           /* 優先レベル11 */
79      D_TMUIPRI_12,           /* 優先レベル12 */
80      D_TMUIPRI_13,           /* 優先レベル13 */
81      D_TMUIPRI_14,           /* 優先レベル14 */
82      D_TMUIPRI_15,           /* 優先レベル15 */
83
84  }T_TMUIntcPri;
85
86  /* P 分割値 */
87  typedef enum {
88
89      D_TMUIDIVValue_4   = 4,          /* P /4 */
90      D_TMUIDIVValue_16  = 16,         /* P /16 */
91      D_TMUIDIVValue_64  = 64,         /* P /64 */
92      D_TMUIDIVValue_256 = 256,        /* P /256 */
93      D_TMUIDIVValue_1024 = 1024,      /* P /1024 */
94
95  }T_TMUPClockDiv;
96
97  /* タイムアウト時コールバック関数 */
98  typedef void (*TMU_TIMEOUT_CALLBACK)(void);
99
100 /* ==== 構造体定義 ==== */
101 /* タイマ生成用構造体 */
102 typedef struct {
103
104     T_TMUClockSelect  mClockSelect; /* クロックセレクト */
105     T_TMUIntcPri      mIntcPri;     /* TMU_割り込み優先レベル */
106
107 } T_TMUTimerTYPE;
108
109 /* TCOR 設定情報構造体 */
110 typedef struct {
111
112     unsigned long    mTimeValue;     /* タイマ設定値 */
113
114 } T_TMUSettCOR_INFO;
115
116 /* TMU 保存情報構造体 */
117 typedef struct {
118
119     T_TMUClockSelect  mClockSelect; /* クロックセレクト */
120     TMU_TIMEOUT_CALLBACK mfunc;     /* タイムアウト時コールバック関数 */
121
122 } T_TMUSave_INFO;
123
124 /* クロックセレクト対応構造体 */
125 typedef struct {
126
127     T_TMUClockSelect  mClockSetting;
128     T_TMUPClockDiv    mClockDivValue;
129

```

```
130     } T_TMU_CLOCK_SELECT;
131
132     /* ==== 関数宣言 ==== */
133     void tmu0_create(T_TMU_TimerTYPE *i_pTimerType);
134     void tmu0_start(T_TMU_SETTCOR_INFO *i_pTCORInfo, TMU_TIMEOUT_CALLBACK i_func);
135     void tmu0_stop(void);
136     void tmu0_compare_match(void);
```

(4) サンプルプログラムリスト"intprg.c"

TMU0 に関連する割り込み処理を実装しています。

```

1      /*
2      以下の INTC_RESPONSEWAIT は、周辺モジュールの優先順位判定時間待ち用の値です。
3      int_responstime_wait() にパラメータ指定して使用ください。
4      int_responstime_wait() については、common.src を参照ください。
5
6      周辺モジュールの優先順位判定時間は、5Pcyc となります。
7      int_responstime_wait() では、Icyc を 3cyc 実行する処理を、指定されたパラメータの回数
8      実行することで wait 処理を行います。
9
10     int_responstime_wait() のパラメータ指定値計算
11     H'0E >= (1/Pcyc*5cyc)/(1/Icyc*3cyc)
12     念のため 1 回多く実行します。
13
14     注)
15     I = 266.66MHz、P = 33.33MHz 動作時の設定となっています。
16     これらを変更する場合はこちらの設定値も見直してください。
17     */
18     #define INTC_RESPONSEWAIT (0x0000000E)
19
20     ...途中省略...
21
22     /* H'400 TMU ch-0 underflow interrupt */
23     void INT_TMU0_TUNIO(void)
24     {
25         /* アンダフローフラグの確認 */
26         if((TMU0.TCR & 0x0100) == 0x0100)
27         {
28             /* アンダフローフラグのクリア */
29             TMU0.TCR &= ~0x0100;
30
31             /* アンダフロー時コール処理 */
32             tmu0_compare_match();
33
34             /* 優先順位判定時間分待ち処理 */
35             int_responstime_wait_for_tmu0();
36         }
37     }
38
39     ...途中省略...
40
41
42     void int_responstime_wait_for_tmu0(void)
43     {
44         unsigned short dummy;
45
46         /* 更新したはずの割り込み要因による割り込みの受け付けを避ける対応 */
47         dummy = TMU0.TCR;
48         int_responstime_wait(INTC_RESPONSEWAIT); /* 優先順位判定時間分待ち */
49     }
    
```

(5) サンプルプログラムリスト"vecttbl.src"

TMU0 割り込み実行時の割り込み優先度を設定しています。

TMU0 割り込みの優先度を 1 に設定しているため、TMU0 割り込み中に新たな TMU0 割り込みが発生しないように優先度に 1 を設定しています。

```

1      ...途中省略...
2
3      ;TMU
4          ;H'400      TMU TUNIO
5          .data.b    H'10
6
7      途中省略...
```

6. 参考ドキュメント

- ソフトウェアマニュアル
SH-4A ソフトウェアマニュアル (RJJ09B0090)
(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。)
- ハードウェアマニュアル
SH7730 グループ ハードウェアマニュアル (RJJ09B0339)
(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。)

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://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2009.07.28	—	初版発行

すべての商標および登録商標は、それぞれの所有者に帰属します。

本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社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弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証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など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の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、あるいはその他軍事事務の目的で使用しないでください。また、輸出に際しては、「外国為替および外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 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などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりましては、事前に弊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、弊社ホームページ (<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。
5. 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のですが、万一本資料の記述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、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独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。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会ください。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないでください。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
 - 1) 生命維持装置。
 - 2) 人体に埋め込み使用するもの。
 - 3) 治療行為（患部切り出し、薬剤投与等）を行うもの。
 - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された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、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて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、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に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。弊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、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）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。
 - 1 1. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
 - 1 2. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断りいたします。
 - 1 3. 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弊社営業窓口までご照会ください。

D039444