

SH7253 SH7256R グループ

R01AN1638JJ0100

Rev.1.00

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

2013.04.02

要旨

本アプリケーションノートは、アドバンスドタイマユニット-III (ATU-III) を用いた、タイマ動作例についてまとめたものです。

ATU-III は、各タイマブロックは、それぞれが異なる機能を備えており、互いに独立して動作することができます。また、クロックバスを介して複数のタイマを連動して動作させることもできます。タイマブロックは、同一の機能を持った 1 個以上のタイマサブブロックによって構成され、各サブブロックはさらに 1 個以上のチャンネルを備えています。

本アプリケーションノートに掲載されているタスク例は動作確認済みですが、実際にご使用になる場合には、必ず動作環境を確認の上ご使用くださいますようお願いいたします。

【注】本アプリケーションノートのサンプルコードは SH7254R グループ用に作成しています。SH7253 グループでご使用の場合は、本文「SH7253 グループでご使用の場合の変更点」に従い、修正してください。SH7256 グループでご使用の場合は、製品レジスタ定義ファイル `iodefine.h` を SH7256R グループ用のものと差替えるのみで構いません。ソース上の各レジスタ名は、`iodefine.h` に合わせ、修正してください。

動作確認デバイス

SH72546R

目次

1.	はじめに	3
1.1	仕様	3
1.2	使用機能	3
1.3	適用条件	3
2.	応用例の説明	3
2.1	使用機能の動作概要	3
2.2	動作例 1 パルスの周期測定[タイマ A]	4
2.3	動作例 2 パルスの High 幅測定[タイマ A]	12
2.4	動作例 3 ノイズキャンセル(後続エッジキャンセル)[タイマ A]	21
2.5	動作例 4 ノイズキャンセル(先行エッジキャンセル)[タイマ A]	29
2.6	動作例 5 イベント周期測定[タイマ C]	37
2.7	動作例 6 パルス出力(トグル出力、タイマカウントクリア)[タイマ C]	45
2.8	動作例 7 パルス出力(トグル出力、コンペア値加算)[タイマ C]	52
2.9	動作例 8 パルス出力(High/Low 切替出力、コンペア値加算)[タイマ C]	60
2.10	動作例 9 ワンショットパルス出力(オフセット付)[タイマ D]	68
2.11	動作例 10 ワンショットパルス出力(オフセット、ターミナ付)[タイマ D]	79
2.12	動作例 11 PWM 波形出力[タイマ E]	90
2.13	動作例 12 有効エッジ入力間隔[タイマ F]	99
2.14	動作例 13 一定時間内エッジカウント[タイマ F]	107
2.15	動作例 14 入力 High/Low 期間計測[タイマ F]	115
2.16	動作例 15 PWM 入力波形計測[タイマ F]	123
2.17	動作例 16 回転速度/パルス計測[タイマ F]	132
2.18	動作例 17 アップダウンイベントカウント[タイマ F]	141
2.19	動作例 18 4 通りイベントカウント[タイマ F]	149
2.20	動作例 19 コンペアマッチ割り込み発生[タイマ G]	157
2.21	動作例 20 コンペアマッチ発生による他モジュールの起動[タイマ G]	163
2.22	動作例 21 コンペアマッチ発生による他モジュールの起動[タイマ G]	171
2.23	動作例 22 コンペアマッチ割り込み発生[タイマ H]	182
2.24	動作例 23 FIFO レジスタへの値の格納[タイマ J]	189
2.25	動作例 24 FIFO レジスタへの値の格納[タイマ J]	199
2.26	動作例 25 倍周クロックの生成[タイマ B]	209
2.27	動作例 26 倍周クロックを用いた入力信号の欠け歯の検出[タイマ B]	222
2.28	動作例 27 倍周補正クロック信号の生成[タイマ B]	232
2.29	動作例 28 倍周補正クロック信号の生成[タイマ B]	245

1. はじめに

1.1 仕様

本アプリケーションノートでは、SH72546R のアドバンスドタイマユニット-III (ATU-III) の使用方法およびファームウェアの作成例を掲載しています。

1.2 使用機能

- アドバンスドタイマユニット-III (ATU-III)
- ダイレクトメモリアクセスコントローラ (DMAC)
- A/D 変換器 (ADC)
- 割り込みコントローラ (INTC)
- ピンファンクションコントローラ (PFC)

1.3 適用条件

- マイコン : SH72546R
- 動作周波数 : 内部クロック(ϕ) 200MHz
周辺クロック(P ϕ) 40MHz
- 統合開発環境 : ルネサスエレクトロニクス(株)
High-performance Embedded Workshop バージョン=V.4.09.00.007
- C コンパイラ : ルネサスエレクトロニクス(株)
SuperH RISC engine family V.9.04 Release 00
- コンパイルオプション : `-cpu=sh2afpu -object="$$(CONFIGDIR)¥$(FILELEAF).obj" -debug -gbr=auto
-chgincpath -errorpath -global_volatile=0 -opt_range=all -infinite_loop=0
-del_vacant_loop=0 -struct_alloc=1 -nologo`

2. 応用例の説明

本応用例では、ATU-IIIのタイマ A~J の各種機能の動作例をそれぞれ纏めたものです。インプットキャプチャ機能などで必要となる外部入力信号は任意に生成する必要が有ります。

2.1 使用機能の動作概要

ATU-III は、タイマ A~J の 9 種類のタイマブロックと、プリスケアラ、および共通制御部から構成されています。各タイマブロックは、それぞれが異なる機能を備えており、互いに独立して動作することができます。また、クロックバスを介して複数のタイマを連動して動作させることもできます。タイマブロックは、同一の機能を持った 1 個以上のタイマサブブロックによって構成され、各サブブロックはさらに 1 個以上のチャンネルを備えています。

2.2 動作例 1 パルスの周期測定[タイマ A]

2.2.1 概要

- 1) 図 2.1に示すようにパルスの周期を測定し、結果を RAM に保存します。
- 2) パルスの周期は次の式で求められます。

[パルスの周期(ns)=タイマ値×Pφ周期(40MHz 動作時 25ns)×2 (プリスケアラの分周比)]

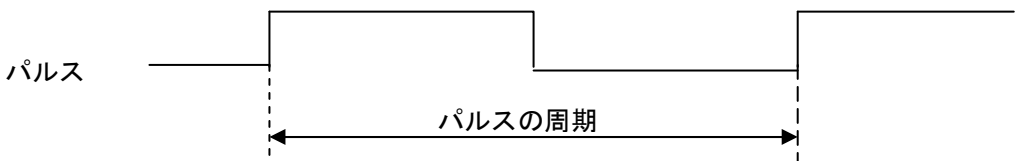


図 2.1 パルス周期測定タイミング

2.2.2 使用機能説明

表 2.1に使用端子及び関連レジスタの機能割り付けを示します。

表 2.1 ATU-III機能割り付け

使用端子		機能
端子	TIA00	測定するパルスをこの端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ A レジスタ	TIOR1A	外部入力のキャプチャするエッジを設定します。
	TIERA	TIA00 のインプットキャプチャ割り込みを許可/禁止を設定します
	TCNTA	選択したクロックバスでタイマカウントを行います。
	TSRA	インプットキャプチャが発生した場合にフラグがセットされます。
PFC レジスタ	PECR1	端子機能を設定します。
INTC レジスタ	IPR06	TIA00 のインプットキャプチャ割り込みの優先度を設定します。

2.2.3 動作説明

図 2.2に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってパルスの周期測定を行います。外部入力パルスの立ち上がりに応じてインプットキャプチャ割り込みによるタイマカウントを開始し現在のキャプチャ値を保存します。次の立ち上がりで再度インプットキャプチャによる割り込みで、前回のキャプチャ値と今回のキャプチャ値との差分を取得することで、パルス周期のカウンタ値を取得します。

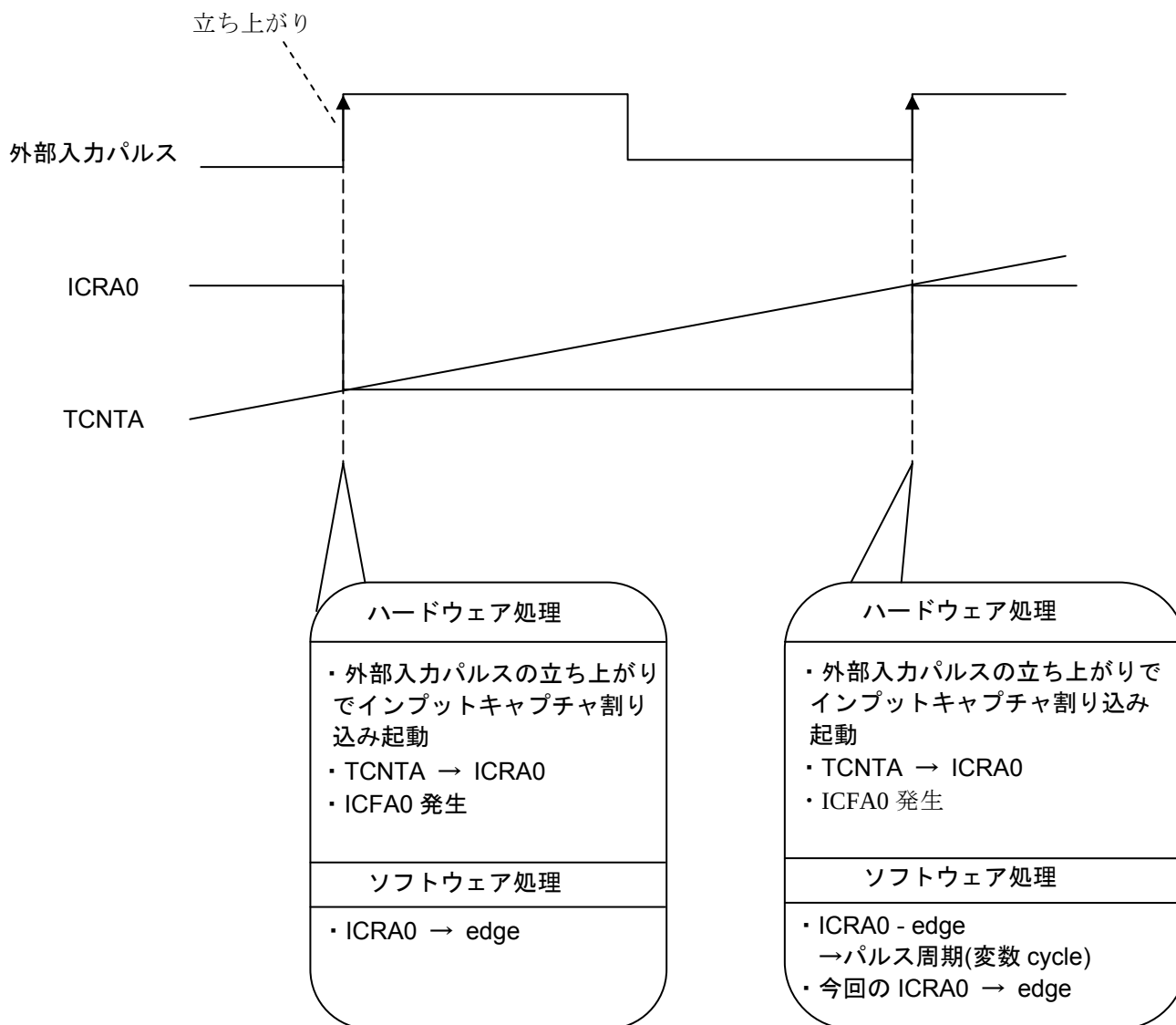


図 2.2 パルス周期計測動作原理

2.2.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
パルスの周期測定	ICIA0	ICFA0 によって起動し、ICRA0 の値によってパルスの周期を測定します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
cycle	パルスの周期に相当するタイマ値を設定します。 パルスの周期は次の式で求められます。 パルスの周期(ns)=cycle の値 × Pφ 周期(40MHz 動作時 25ns) × 2 (プリスケアラの分周比)	unsigned long	パルスの周期測定
edge	インプットキャプチャ発生時の TCNT0 の値を保存します。		

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE0 端子を入力端子 TIA00 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 2 に設定します。	0x0001	
ATUENR	ATU-III タイマ A 及びプリスケアラのカウント動作を設定します。	0x0003	
TIOR1A	TIA00 からの入力信号の立ち上がりエッジでインプットキャプチャをするように設定します。	0x0001	
TIERA	TIA00 のインプットキャプチャ割り込み要求を許可します。	0x01	
ICRA0	測定パルスのエッジ検出時の TCNTA の値を格納します。	0x0000	
IPR06	TIA00 のインプットキャプチャ割り込み要求の優先度を設定します。	0xF000	
TSRA	タイマ A のフラグをクリアします。	0x00	パルスの周期測定
ICRA0	インプットキャプチャ信号検出時に、TCNTA0 の値を格納します。	-	

2.2.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH725xEVB04-C
32 * Description    : This is the main tutorial code.
33 * Operation      : Measurement of Pulse period.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 25.08.2011 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */

```

```

45  /*****
46  Private global variables and functions
47  *****/
48  unsigned long cycle, edge;
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments     : none
55  * Return Value  : none
56  *****/
57
58  void main(void)
59  {
60  cycle = edge = 0; /* 変数初期化 */
61
62  /* ポートの設定 */
63  /* Configure PECR1
64  b15:14 PE7MD = 0 PE7 入出力 (ポート)
65  b13:12 PE6MD = 0 PE6 入出力 (ポート)
66  b11 Reserved
67  b10 PE5MD = 0 PE5 入出力 (ポート)
68  b9 Reserved
69  b8 PE4MD = 0 PE4 入出力 (ポート)
70  b7 Reserved
71  b6 PE3MD = 0 PE3 入出力 (ポート)
72  b5:4 PE2MD = 0 PE2 入出力 (ポート)
73  b3:2 PE1MD = 0 PE1 入出力 (ポート)
74  b1 Reserved
75  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
76  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
77
78  /* ATU-III の設定 */
79  /* Configure TIOR1A
80  b15:12 Reserved
81  b11:10 IOA5 = 00 TIA のインプットキャプチャ禁止
82  b9:8 IOA4 = 00 TIA のインプットキャプチャ禁止
83  b7:6 IOA3 = 00 TIA のインプットキャプチャ禁止
84  b5:4 IOA2 = 00 TIA のインプットキャプチャ禁止
85  b3:2 IOA1 = 00 TIA のインプットキャプチャ禁止
86  b1:0 IOA0 = 01 TIA の立ち上がりで ICRA にキャプチャ */
87  ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立上りエッジでインプットキャプチャ */
88
89  /* Configure TSRA
90  b7 OVFA = 0 インプットキャプチャなし
91  b6 Reserved
92  b5 ICFA5 = 0 インプットキャプチャなし
93  b4 ICFA4 = 0 インプットキャプチャなし

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94  b3 ICFA3 = 0 インพุットキャプチャなし
95  b2 ICFA2 = 0 インพุットキャプチャなし
96  b1 ICFA1 = 0 インพุットキャプチャなし
97  b0 ICFA0 = 0 インพุットキャプチャなし */
98  ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
99
100 /* Configure TIERA
101 b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
102 b6 Reserved
103 b5 ICEA5 = 0 インพุットキャプチャ割り込み A5 要求の出力を禁止
104 b4 ICEA4 = 0 インพุットキャプチャ割り込み A4 要求の出力を禁止
105 b3 ICEA3 = 0 インพุットキャプチャ割り込み A3 要求の出力を禁止
106 b2 ICEA2 = 0 インพุットキャプチャ割り込み A2 要求の出力を禁止
107 b1 ICEA1 = 0 インพุットキャプチャ割り込み A1 要求の出力を禁止
108 b0 ICEA0 = 1 インพุットキャプチャ割り込み A0 要求の出力を許可 */
109 ATUA.TIERA.BYTE = 0x01; /* TIA00 のインพุットキャプチャ割り込みを許可 */
110
111 /* Configure PSCR0
112 b15:10 Reserved
113 b9:0 PSCn = H'1 プリスケーラの分周比を 2(設定値は-1)に設定 */
114 ATUCTRL.PSCR0.WORD = 0x0001; /* プリスケーラの分周比を 2 に設定 */
115
116 /* 割り込み設定 */
117 /* Configure IPR06
118 b15:12 ATU-A(ICIA0,1) = H'F 対応する割り込みの優先順位を 15 に設定
119 b11:8 ATU-A(ICIA2,3) = H'0 対応する割り込みの優先順位を 0 に設定
120 b7:4 ATU-A(ICIA4,5) = H'0 対応する割り込みの優先順位を 0 に設定
121 b3:0 ATU-A(OVIA) = H'0 対応する割り込みの優先順位を 0 に設定 */
122 INTC.IPR06.WORD = 0xF000; /* TIA00 の割り込み(ICIA0)の優先順位を 15 に設定 */
123
124 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
125
126 /* Configure ATUENR
127 b15:10 Reserved
128 b9 TJE = 0 タイマ J のカウント動作を停止
129 b8 THE = 0 タイマ H のカウント動作を停止
130 b7 TGE = 0 タイマ G のカウント動作を停止
131 b6 TFE = 0 タイマ F のカウント動作を停止
132 b5 TEE = 0 タイマ E のカウント動作を停止
133 b4 TDE = 0 タイマ D のカウント動作を停止
134 b3 TCE = 0 タイマ C のカウント動作を停止
135 b2 TBE = 0 タイマ B のカウント動作を停止
136 b1 TAE = 1 タイマ A のカウント動作を許可
137 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
138 ATUCTRL.ATUENR.WORD = 0x0003; /* タイマ A およびプリスケーラのカウントをスタート */
139
140 while(1); /* 無限ループで待機 */
141 } /* End of function main() */
142
143 /*****

```

```
144  * Function Name : ICIA0
145  * Description   : インพุットキャプチャ割り込み
146  * Arguments     : none
147  * Return Value  : none
148  *****/
149  void ICIA0(void)
150  {
151  /* Configure TSRA
152  b7 OVFA = 0 インพุットキャプチャフラグをクリア
153  b6 Reserved
154  b5 ICFA5 = 0 インพุットキャプチャフラグをクリア
155  b4 ICFA4 = 0 インพุットキャプチャフラグをクリア
156  b3 ICFA3 = 0 インพุットキャプチャフラグをクリア
157  b2 ICFA2 = 0 インพุットキャプチャフラグをクリア
158  b1 ICFA1 = 0 インพุットキャプチャフラグをクリア
159  b0 ICFA0 = 0 インพุットキャプチャフラグをクリア */
160  ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
161  cycle = ATUA.ICRA0 - edge; /* パルス周期演算 */
162  edge = ATUA.ICRA0; /* キャプチャ値保存 */
163  } /* End of function ICIA0() */
```

2.2.6 SH7253 グループでご使用の場合の変更点

本文 2.2 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.2.5 サンプルプログラムリストの 77 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.3 動作例 2 パルスの High 幅測定[タイマ A]

2.3.1 概要

- 1) 図 2.3に示すようにパルスの周期を測定し、結果を RAM に保存します。
- 2) パルスの周期は次の式で求められます。

[パルスの High 幅(ns)=タイマ値×Pφ周期(40MHz 動作時 25ns)×2 (プリスケアラの分周比)]

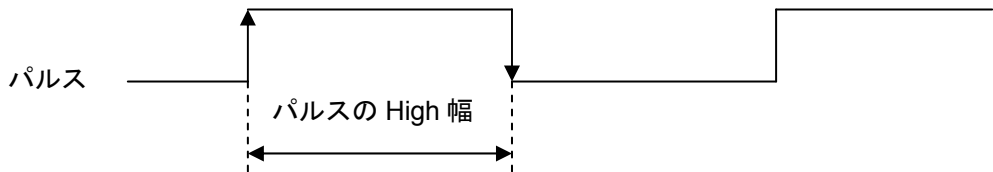


図 2.3 パルスの High 幅計測

2.3.2 使用機能説明

表 2.2に使用端子及び関連レジスタの機能割り付けを示します。

表 2.2 ATU-III機能割り付け

使用端子		機能
端子	TIA00	測定するパルスをこの端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ A レジスタ	TIOR1A	外部入力のキャプチャ対象エッジを設定します。
	TIERA	TIA00 のインプットキャプチャ割り込みを許可/禁止を設定します
	TCNTA	選択したクロックバスでタイマカウントを行います。
	TSRA	インプットキャプチャが発生した場合にフラグがセットされます。
PFC レジスタ	PECR1	端子機能を設定します。
INTC レジスタ	IPR06	TIA00 のインプットキャプチャ割り込みの優先度を設定します。

2.3.3 動作説明

図 2.4に動作原理を示します。これに従って SH72546 のハードウェアおよびソフトウェア処理によってパルスの周期測定を行います。外部入力パルスの立ち上がりに応じてインプットキャプチャ割り込みによるタイマカウントを開始し現在のキャプチャ値を保存します。次は立ち下がりでインプットキャプチャによる割り込みを行います。立ち上がり時と立ち下がり時のキャプチャ値の差分を取得することで、パルス High 幅のカウンタ値を取得します。

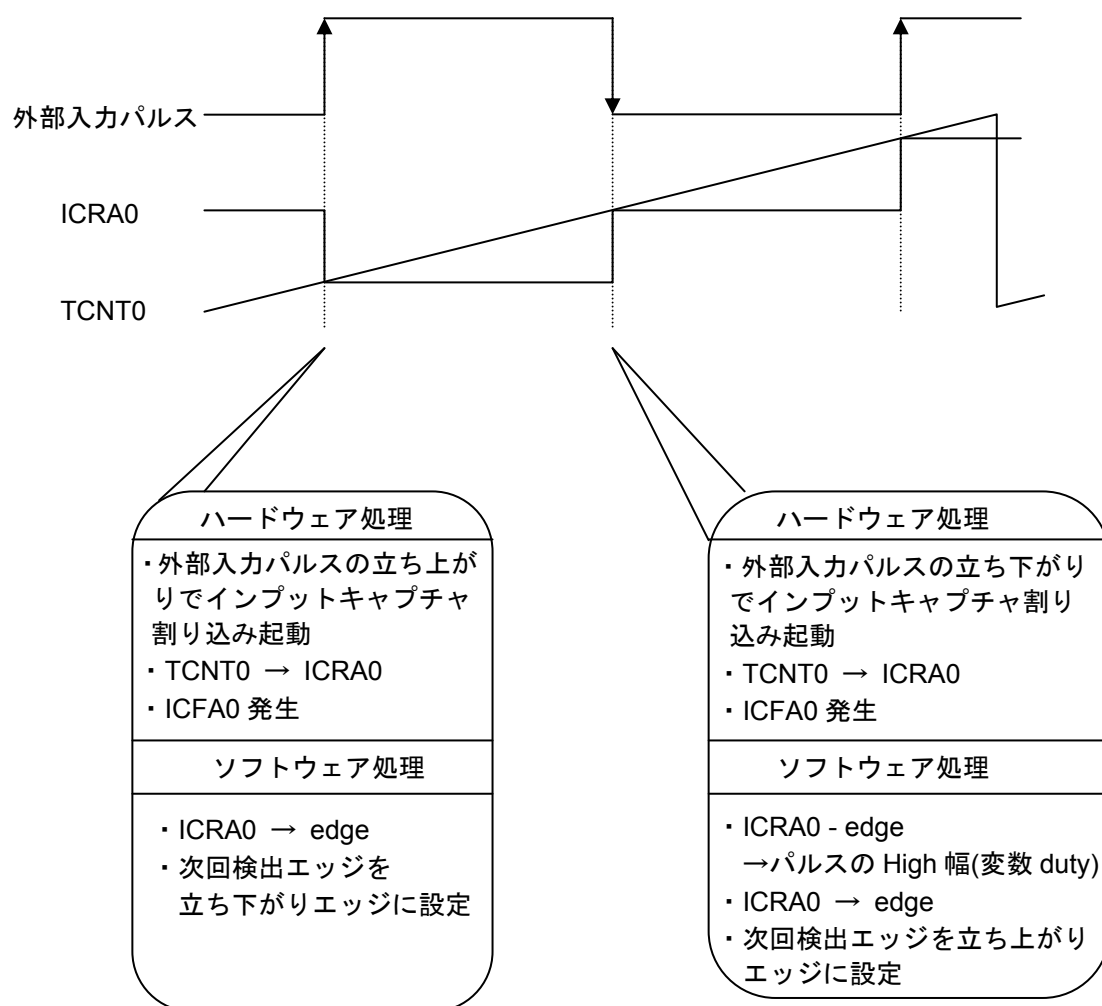


図 2.4 パルスの High 幅計測動作原理

2.3.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
パルスの High 幅測定	ICIA0	ICFA0 によって起動し、ICRA0 の値からパルスの幅を測定します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
duty	パルスの High 幅に相当するタイマ値を設定します。 パルスの High 幅は次の式で求められます。 パルスの High 幅(ns)=duty の値 × Pφ 周期(40MHz 動作時 25ns) × 2 (プリスケアラの分周比)	unsigned long	パルスの周期測定
edge	インプットキャプチャ値を保存します。		

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE0 端子を入力端子 TIA00 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 2 に設定します。	0x0001	
ATUENR	ATU-III チャンネル A 及びプリスケアラのカウント動作を設定します。	0x0003	
TIOR1A	TIA00 からの入力信号の立ち上がりエッジでインプットキャプチャをするように設定します。	0x0001	
	TIA00 からの入力信号の立ち下がりエッジでインプットキャプチャをするように設定します。	0x0002	
TIERA	TIA00 のインプットキャプチャ割り込み要求を許可します。	0x01	
ICRA0	測定パルスのエッジ検出時の TCNTA の値を格納します。	0x0000	
IPR06	TIA00 のインプットキャプチャ割り込み要求の優先度を設定します。	0xF000	パルスの HIGH 幅測定
TSRA	タイマ A のフラグをクリアします。	0x00	
TIOR1A	TIA00 からの入力信号の立ち上がり/立ち下がりインプットキャプチャを行うように設定します。 前回の TIOR1A の値で変わります。	0x0001 0x0002	

2.3.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH725xEVB04-C
32 * Description     : This is the main tutorial code.
33 * Operation      : Measurement of Pulse high width.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 23.12.2011 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long duty, edge;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  duty = edge = 0; /* 変数初期化 */
62
63  /* ポートの設定 */
64  /* Configure PECR1
65  b15:14 PE7MD = 0 PE7 入出力 (ポート)
66  b13:12 PE6MD = 0 PE6 入出力 (ポート)
67  b11 Reserved
68  b10 PE5MD = 0 PE5 入出力 (ポート)
69  b9 Reserved
70  b8 PE4MD = 0 PE4 入出力 (ポート)
71  b7 Reserved
72  b6 PE3MD = 0 PE3 入出力 (ポート)
73  b5:4 PE2MD = 0 PE2 入出力 (ポート)
74  b3:2 PE1MD = 0 PE1 入出力 (ポート)
75  b1 Reserved
76  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
77  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
78
79  /* ATU-III の設定 */
80  /* Configure TIOR1A
81  b15:12 Reserved
82  b11:10 IOA5 = 0 TIA のインプットキャプチャ禁止
83  b9:8 IOA4 = 0 TIA のインプットキャプチャ禁止
84  b7:6 IOA3 = 0 TIA のインプットキャプチャ禁止
85  b5:4 IOA2 = 0 TIA のインプットキャプチャ禁止
86  b3:2 IOA1 = 0 TIA のインプットキャプチャ禁止
87  b1:0 IOA0 = 1 TIA の立ち上がりで ICRA にキャプチャ */
88  ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立ち上がりエッジで ICRA にキャプチャ */
89
90  /* Configure TSRA
91  b7 OVFA = 0 インプットキャプチャフラグをクリア
92  b6 Reserved
93  b5 ICFA5 = 0 インプットキャプチャフラグをクリア
94  b4 ICFA4 = 0 インプットキャプチャフラグをクリア

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

95    b3 ICFA3 = 0 インพุットキャプチャフラグをクリア
96    b2 ICFA2 = 0 インพุットキャプチャフラグをクリア
97    b1 ICFA1 = 0 インพุットキャプチャフラグをクリア
98    b0 ICFA0 = 0 インพุットキャプチャフラグをクリア */
99    ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
100
101    /* Configure TIERA
102    b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
103    b6 Reserved
104    b5 ICEA5 = 0 インพุットキャプチャ割り込み A5 要求の出力を禁止
105    b4 ICEA4 = 0 インพุットキャプチャ割り込み A4 要求の出力を禁止
106    b3 ICEA3 = 0 インพุットキャプチャ割り込み A3 要求の出力を禁止
107    b2 ICEA2 = 0 インพุットキャプチャ割り込み A2 要求の出力を禁止
108    b1 ICEA1 = 0 インพุットキャプチャ割り込み A1 要求の出力を禁止
109    b0 ICEA0 = 1 インพุットキャプチャ割り込み A0 要求の出力を許可 */
110    ATUA.TIERA.BYTE = 0x01; /* TIA00 のインพุットキャプチャ割り込みを許可 */
111
112    /* Configure PSCR0
113    b15:10 Reserved
114    b9:0 PSCn = H'1 プリスケーラの分周比を 2(設定値は-1)に設定 */
115    ATUCTRL.PSCR0.WORD = 0x0001; /* プリスケーラの分周比を 2 に設定 */
116
117    /* 割り込み設定 */
118    /* Configure IPR06
119    b15:12 ATU-A(ICIA0,1) = H'F 対応する割り込みの優先順位を 15 に設定
120    b11:8 ATU-A(ICIA2,3) = H'0 対応する割り込みの優先順位を 0 に設定
121    b7:4 ATU-A(ICIA4,5) = H'0 対応する割り込みの優先順位を 0 に設定
122    b3:0 ATU-A(OVIA) = H'0 対応する割り込みの優先順位を 0 に設定 */
123    INTC.IPR06.WORD = 0xF000; /* TIA00 の割り込み(ICIA0)の優先順位を 15 に設定 */
124
125    set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
126
127    /* Configure ATUENR
128    b15:10 Reserved
129    b9 TJE = 0 タイマ J のカウント動作を停止
130    b8 THE = 0 タイマ H のカウント動作を停止
131    b7 TGE = 0 タイマ G のカウント動作を停止
132    b6 TFE = 0 タイマ F のカウント動作を停止
133    b5 TEE = 0 タイマ E のカウント動作を停止
134    b4 TDE = 0 タイマ D のカウント動作を停止
135    b3 TCE = 0 タイマ C のカウント動作を停止
136    b2 TBE = 0 タイマ B のカウント動作を停止
137    b1 TAE = 1 タイマ A のカウント動作を許可
138    b0 PSCE = 1 プリスケーラのカウント動作を許可 */
139    ATUCTRL.ATUENR.WORD = 0x0003; /* タイマ A およびプリスケーラのカウントをスタート */
140
141    while(1); /* 無限ループで待機 */
142    } /* End of function main() */
143
144    /*****

```

```

145  * Function Name : ICIA0
146  * Description   : インพุットキャプチャ割り込み
147  * Arguments     : none
148  * Return Value  : none
149  *****/
150 void ICIA0(void) /* インพุットキャプチャ割り込み */
151 {
152     /* Configure TSRA
153     b7 OVFA = 0 インพุットキャプチャフラグをクリア
154     b6 Reserved
155     b5 ICFA5 = 0 インพุットキャプチャフラグをクリア
156     b4 ICFA4 = 0 インพุットキャプチャフラグをクリア
157     b3 ICFA3 = 0 インพุットキャプチャフラグをクリア
158     b2 ICFA2 = 0 インพุットキャプチャフラグをクリア
159     b1 ICFA1 = 0 インพุットキャプチャフラグをクリア
160     b0 ICFA0 = 0 インพุットキャプチャフラグをクリア */
161     ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
162
163     /* Configure TI0R1A
164     b15:12 Reserved
165     b11:10 IOA5 = 00 TIA のインพุットキャプチャ禁止
166     b9:8 IOA4 = 00 TIA のインพุットキャプチャ禁止
167     b7:6 IOA3 = 00 TIA のインพุットキャプチャ禁止
168     b5:4 IOA2 = 00 TIA のインพุットキャプチャ禁止
169     b3:2 IOA1 = 00 TIA のインพุットキャプチャ禁止
170     b1:0 IOA0 = 01 TIA の立ち上がりで ICRA にキャプチャ */
171     if(ATUA.TI0R1A.WORD == 0x0001) /* 今回読み込んだ値が立上がりエッジならば */
172     {
173         edge = ATUA.ICRA0; /* インพุットキャプチャが発生した時点でのカウンツの値を保存 */
174
175         /* Configure TI0R1A
176         b15:12 Reserved
177         b11:10 IOA5 = 00 TIA のインพุットキャプチャ禁止
178         b9:8 IOA4 = 00 TIA のインพุットキャプチャ禁止
179         b7:6 IOA3 = 00 TIA のインพุットキャプチャ禁止
180         b5:4 IOA2 = 00 TIA のインพุットキャプチャ禁止
181         b3:2 IOA1 = 00 TIA のインพุットキャプチャ禁止
182         b1:0 IOA0 = 10 TIA の立ち下がりがりで ICRA にキャプチャ */
183         ATUA.TI0R1A.WORD = 0x0002; /* 次回読み込むのは立下がりエッジ */
184     }
185     else /* 今回読み込んだ値が立下がりエッジならば */
186     {
187         duty = ATUA.ICRA0 - edge; /* 今回のインพุットキャプチャ時のカウンツの値から
188         前回のインพุットキャプチャ時の値を引き、Low 期間の値を求める */
189         /* Configure TI0R1A
190         b15:12 Reserved
191         b11:10 IOA5 = 00 TIA のインพุットキャプチャ禁止
192         b9:8 IOA4 = 00 TIA のインพุットキャプチャ禁止
193         b7:6 IOA3 = 00 TIA のインพุットキャプチャ禁止
194         b5:4 IOA2 = 00 TIA のインพุットキャプチャ禁止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```
195    b3:2 IOA1 = 00 TIA のインプットキャプチャ禁止
196    b1:0 IOA0 = 01 TIA の立ち上がりで ICRA にキャプチャ */
197    ATUA.TIOR1A.WORD = 0x0001; /* 次回読み込むのは立ち上がりエッジ */
198    }
199    } /* End of function ICIA0() */
```

2.3.6 SH7253 グループでご使用の場合の変更点

本文 2.3 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.3.5 サンプルプログラムリストの 77 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.4 動作例 3 ノイズキャンセル(後続エッジキャンセル)[タイマ A]

2.4.1 概要

図 2.5に示すように後続エッジキャンセル機能は、入力される信号に混ざるノイズを除去する為に、入力から一定期間内のレベル変化を無視する機能です。タイマとのコンペアマッチでノイズキャンセル期間を制御します。

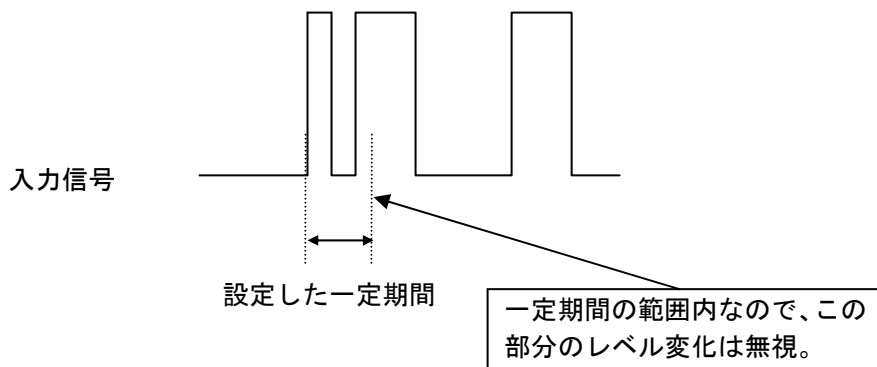


図 2.5 後続ノイズキャンセルの概念図

2.4.2 使用機能説明

表 2.3に使用端子及び関連レジスタの機能割り付けを示します。

表 2.3 ATU-III機能割り付け

使用端子		機能
端子	TIA00	測定する入力信号をこの端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	NCMR	各タイマのノイズキャンセル動作モードおよびカウントクロックを設定します。
タイマ A レジスタ	TIOR1A	外部入力のキャプチャするエッジを設定します。
	TIOR2A	ノイズキャンセルに関する内容を設定します。
	NCNTA	エッジ入力と同時にカウントを開始し、ノイズキャンセルの期間を測定します。
	NCRA0	ノイズキャンセルの期間を設定します。
PFC レジスタ	PECR1	端子機能を設定します。

2.4.3 動作説明

図 2.6に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって後続ノイズキャンセルを行います。ノイズキャンセル後の信号は、タイマカウントの開始後はコンペアマッチが発生するまでは更新されません。（図は立ち上がりエッジを検出する場合の例）

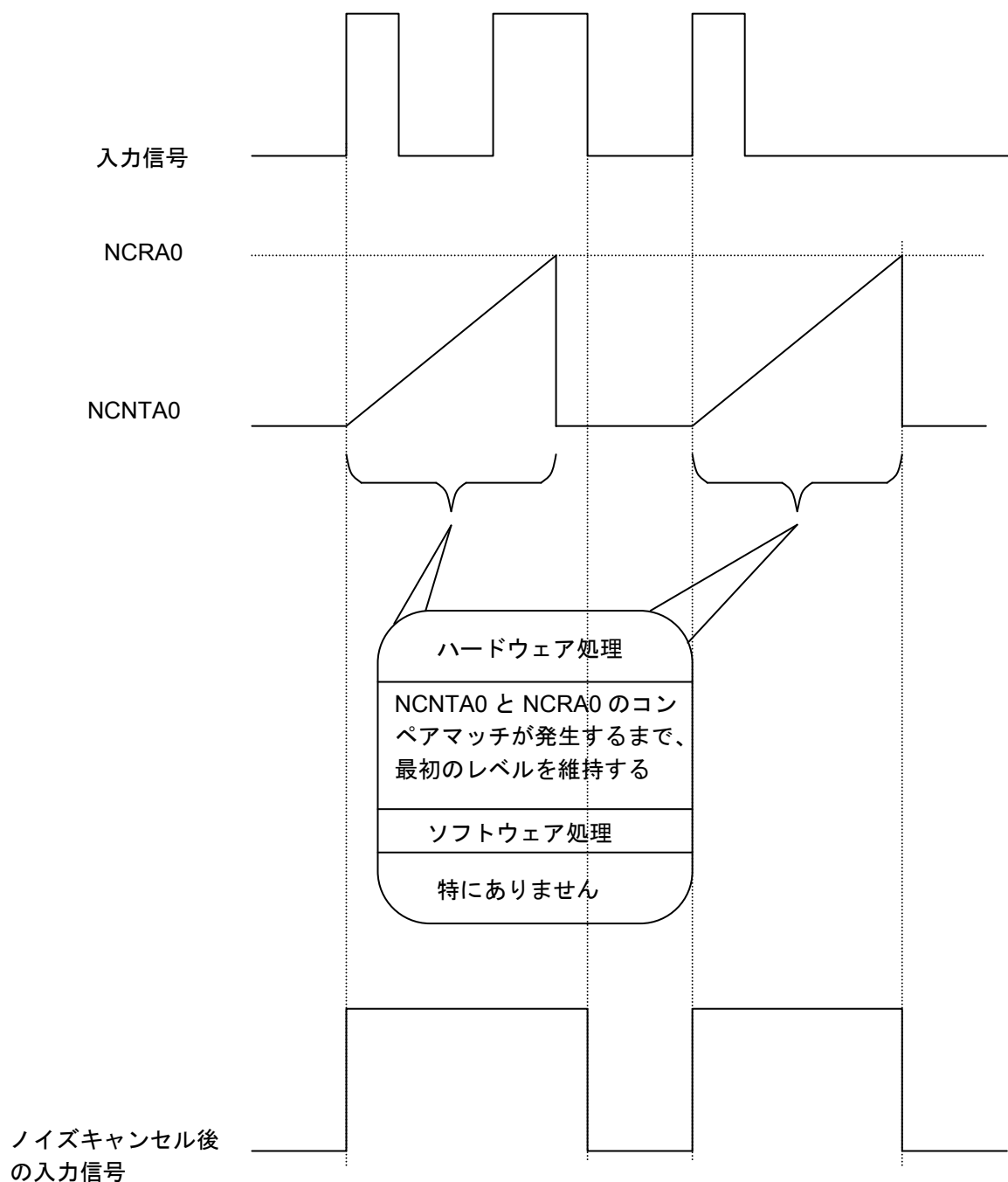


図 2.6 後続ノイズキャンセル動作原理

2.4.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。

● 使用変数の説明

本タスクでは変数は使用していません。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE0 端子を入力端子 TIA00 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 2 に設定します。	0x0001	
ATUENR	ATU-IIIチャンネル A 及びプリスケアラのカウント動作を設定します。	0x0003	
NCMR	タイマ A のノイズキャンセル機能を後続ノイズキャンセルに設定します。	0x00	
TIOR1A	TIA00 からの入力信号の立ち上がりエッジでインプットキャプチャをするように設定します。	0x0001	
TIOR2A	ノイズキャンセルカウンタのクロックバス及び動作許可を設定します。	0x0001	
NCRA0	ノイズキャンセルカウンタの上限を 0x12 に設定します。	0x12	

2.4.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH725xEVB04-C
32 * Description    : This is the main tutorial code.
33 * Operation      : Subsequent edge noise cancellation.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 23.12.2011 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments    : none
55  * Return Value  : none
56  *****/
57  void main(void)
58  {
59  /* ポートの設定 */
60  /* Configure PECR1
61  b15:14 PE7MD = 0 PE7 入出力 (ポート)
62  b13:12 PE6MD = 0 PE6 入出力 (ポート)
63  b11 Reserved
64  b10 PE5MD = 0 PE5 入出力 (ポート)
65  b9 Reserved
66  b8 PE4MD = 0 PE4 入出力 (ポート)
67  b7 Reserved
68  b6 PE3MD = 0 PE3 入出力 (ポート)
69  b5:4 PE2MD = 0 PE2 入出力 (ポート)
70  b3:2 PE1MD = 0 PE1 入出力 (ポート)
71  b1 Reserved
72  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
73  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
74
75  /* ATU-III の設定 */
76  /* Configure TIOR1A
77  b15:12 Reserved
78  b11:10 IOA5 = 0 TIA のインプットキャプチャ禁止
79  b9:8 IOA4 = 0 TIA のインプットキャプチャ禁止
80  b7:6 IOA3 = 0 TIA のインプットキャプチャ禁止
81  b5:4 IOA2 = 0 TIA のインプットキャプチャ禁止
82  b3:2 IOA1 = 0 TIA のインプットキャプチャ禁止
83  b1:0 IOA0 = 1 TIA の立ち上がりで ICRA にキャプチャ */
84  ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立ち上がりエッジで ICRA にキャプチャ */
85  /* Configure TSRA
86  b7 OVFA = 0 インプットキャプチャフラグをクリア
87  b6 Reserved
88  b5 ICFA5 = 0 インプットキャプチャフラグをクリア
89  b4 ICFA4 = 0 インプットキャプチャフラグをクリア
90  b3 ICFA3 = 0 インプットキャプチャフラグをクリア
91  b2 ICFA2 = 0 インプットキャプチャフラグをクリア
92  b1 ICFA1 = 0 インプットキャプチャフラグをクリア
93  b0 ICFA0 = 0 インプットキャプチャフラグをクリア */
94  ATUA.TSRA.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */

```

```
95
96 /* Configure NCMR
97 b7 NCCSEL = 0 内部周辺クロックの 128 分周でカウントアップ
98 b6:4 Reserved
99 b3 NCMJ = 0 タイマ J 後続エッジキャンセルモード
100 b2 NCMF = 0 タイマ F 後続エッジキャンセルモード
101 b1 NCMC = 0 タイマ C 後続エッジキャンセルモード
102 b0 NCMA = 0 タイマ A 後続エッジキャンセルモード */
103 ATUCTRL.NCMR.BYTE = 0x00; /* タイマ A のノイズキャンセラを後続エッジキャンセルに設定 */
104
105 /* Configure TIOR2A
106 b15:14 Reserved
107 b13 NCKA5 = 0 NCNTA5 のカウントソースクロックにノイズキャンセラカウントクロックを選択
108 b12 NCKA4 = 0 NCNTA4 のカウントソースクロックにノイズキャンセラカウントクロックを選択
109 b11 NCKA3 = 0 NCNTA3 のカウントソースクロックにノイズキャンセラカウントクロックを選択
110 b10 NCKA2 = 0 NCNTA2 のカウントソースクロックにノイズキャンセラカウントクロックを選択
111 b9 NCKA1 = 0 NCNTA1 のカウントソースクロックにノイズキャンセラカウントクロックを選択
112 b8 NCKA0 = 0 NCNTA0 のカウントソースクロックにノイズキャンセラカウントクロックを選択
113 b7:6 Reserved
114 b5 NCEA5 = 0 TIA05 入力のノイズキャンセル機能を無効
115 b4 NCEA4 = 0 TIA04 入力のノイズキャンセル機能を無効
116 b3 NCEA3 = 0 TIA03 入力のノイズキャンセル機能を無効
117 b2 NCEA2 = 0 TIA02 入力のノイズキャンセル機能を無効
118 b1 NCEA1 = 0 TIA01 入力のノイズキャンセル機能を無効
119 b0 NCEA0 = 1 TIA00 入力のノイズキャンセル機能を有効 */
120 ATUA.TIOR2A.WORD = 0x0001; /* TIA00 のノイズキャンセラを有効に設定
121     カウンタのクロックをクロックバス 0 に設定 */
122 /* Configure NCRA0
123 b7:0 NCCNTAn = H'12 ノイズキャンセルタイムの上限値設定 */
124 ATUA.NCRA0 = 0x12; /* TIA00 のノイズキャンセルカウンタの上限を 18 に設定 */
125
126 /* Configure PSCR0
127 b15:10 Reserved
128 b9:0 PSCn = H'1 プリスケーラの分周比を 2(設定値は-1)に設定 */
129 ATUCTRL.PSCR0.WORD = 0x0001; /* プリスケーラの分周比を 2 に設定 */
130
131 /* Configure ATUENR
132 b15:10 Reserved
133 b9 TJE = 0 タイマ J のカウント動作を停止
134 b8 THE = 0 タイマ H のカウント動作を停止
135 b7 TGE = 0 タイマ G のカウント動作を停止
136 b6 TFE = 0 タイマ F のカウント動作を停止
137 b5 TEE = 0 タイマ E のカウント動作を停止
138 b4 TDE = 0 タイマ D のカウント動作を停止
139 b3 TCE = 0 タイマ C のカウント動作を停止
140 b2 TBE = 0 タイマ B のカウント動作を停止
141 b1 TAE = 1 タイマ A のカウント動作を許可
142 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
143 ATUCTRL.ATUENR.WORD = 0x0003; /* タイマ A およびプリスケーラのカウントをスタート */
144
```

```
145    while(1);  
146    } /* End of function main() */
```

2.4.6 SH7253 グループでご使用の場合の変更点

本文 2.4 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.4.5 サンプルプログラムリストの 73 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.5 動作例 4 ノイズキャンセル(先行エッジキャンセル)[タイマ A]

2.5.1 概要

図 2.7に示すように先行エッジキャンセル機能は、入力される信号に混ざるノイズを除去するために、期間が一定未満のレベル変化を無視する機能です。

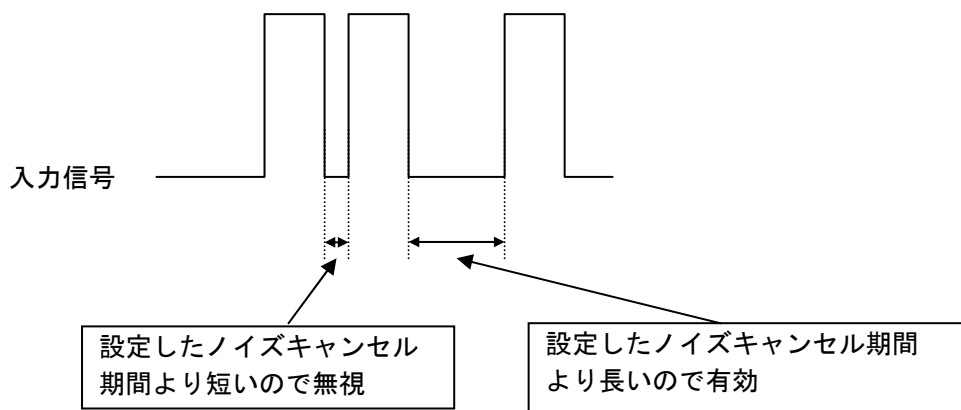


図 2.7 ノイズキャンセルの概念図

2.5.2 使用機能説明

表 2.4に使用端子及び関連レジスタの機能割り付けを示します。

表 2.4 ATU-III機能割り付け

使用端子		機能
端子	TIA0	測定する入力信号をこの端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	NCMR	各タイマのノイズキャンセル動作モードおよびカウントクロックを設定します。
タイマ A レジスタ	TIOR1A	外部入力のキャプチャするエッジを設定します。
	TIOR2A	ノイズキャンセルに関する内容を設定します。
	NCNTA	エッジ入力と同時にカウントを開始し、ノイズキャンセルの期間を測定します。
	NCRA0	ノイズキャンセルの期間を設定します。

2.5.3 動作説明

図 2.8に動作原理を示します。信号レベルが指定期間以上 High でないものをノイズと見なし無視する為に、コンペアマッチ発生後の入力信号をノイズキャンセル済の信号として取り込みます。

これに従って SH72546R のハードウェアおよびソフトウェア処理によって先行ノイズキャンセル動作を行います。（図は立ち上がりエッジを検出する場合の例）

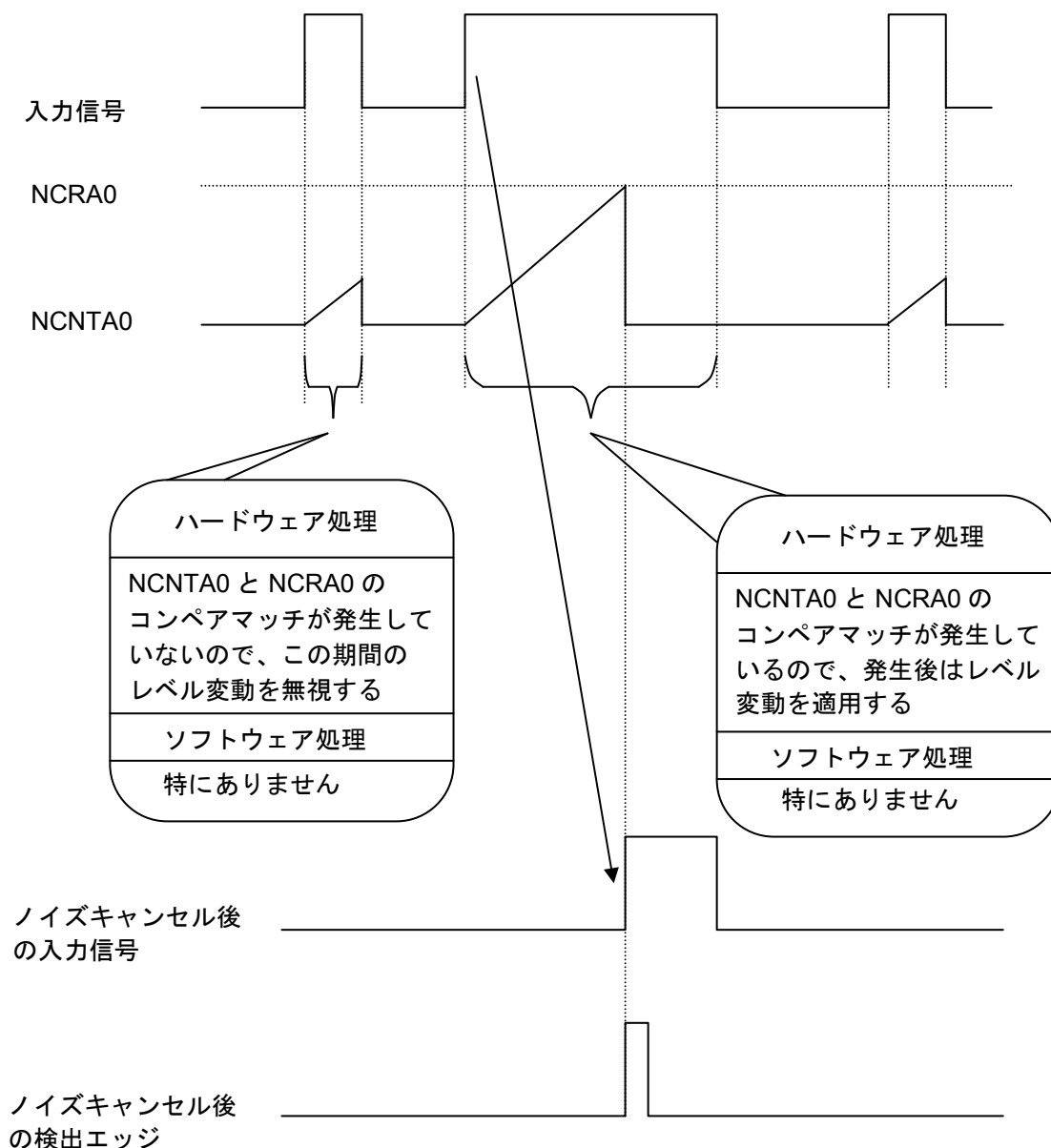


図 2.8 先行ノイズキャンセル動作原理

2.5.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。

- 使用変数の説明

本タスクでは変数は使用していません。

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE0 端子を入力端子 TIA00 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 2 に設定します。	0x0001	
ATUENR	ATU-IIIチャンネル A 及びプリスケアラのカウント動作を設定します。	0x0003	
NCMR	タイマ A のノイズキャンセル機能を先行ノイズキャンセルに設定します。	0x01	
TIOR1A	TIA00 からの入力信号の立ち上がりエッジでインプットキャプチャをするように設定します。※	0x0001	
TIOR2A	ノイズキャンセルカウンタのクロックバス及び動作許可を設定します。	0x0001	
NCRA0	ノイズキャンセルカウンタの上限を 0x12 に設定します	0x12	

※キャプチャ後のレベル変動が監視されるので、このレジスタでノイズキャンセルの対象になるレベルを設定します。

2.5.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Leading-edge noise cancellation.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments    : none
55  * Return Value  : none
56  *****/
57
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PECR1
63  b15:14 PE7MD = 0 PE7 入出力 (ポート)
64  b13:12 PE6MD = 0 PE6 入出力 (ポート)
65  b11 Reserved
66  b10 PE5MD = 0 PE5 入出力 (ポート)
67  b9 Reserved
68  b8 PE4MD = 0 PE4 入出力 (ポート)
69  b7 Reserved
70  b6 PE3MD = 0 PE3 入出力 (ポート)
71  b5:4 PE2MD = 0 PE2 入出力 (ポート)
72  b3:2 PE1MD = 0 PE1 入出力 (ポート)
73  b1 Reserved
74  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
75  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
76
77  /* ATU-III の設定 */
78  /* Configure TIOR1A
79  b15:12 Reserved
80  b11:10 IOA5 = 0 TIA のインプットキャプチャ禁止
81  b9:8 IOA4 = 0 TIA のインプットキャプチャ禁止
82  b7:6 IOA3 = 0 TIA のインプットキャプチャ禁止
83  b5:4 IOA2 = 0 TIA のインプットキャプチャ禁止
84  b3:2 IOA1 = 0 TIA のインプットキャプチャ禁止
85  b1:0 IOA0 = 1 TIA の立ち上がりで ICRA にキャプチャ */
86  ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立ち上がりエッジで ICRA にキャプチャ */
87  /* Configure TSRA
88  b7 OVFA = 0 インプットキャプチャフラグをクリア
89  b6 Reserved
90  b5 ICFA5 = 0 インプットキャプチャフラグをクリア
91  b4 ICFA4 = 0 インプットキャプチャフラグをクリア
92  b3 ICFA3 = 0 インプットキャプチャフラグをクリア
93  b2 ICFA2 = 0 インプットキャプチャフラグをクリア
94  b1 ICFA1 = 0 インプットキャプチャフラグをクリア

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

95  b0 ICFA0 = 0 インพุットキャプチャフラグをクリア */
96  ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
97
98  /* Configure NCMR
99  b7 NCCSEL = 0 内部周辺クロックの 128 分周でカウントアップ
100 b6:4 Reserved
101 b3 NCMJ = 0 タイマ J 後続エッジキャンセルモード
102 b2 NCMF = 0 タイマ F 後続エッジキャンセルモード
103 b1 NCMC = 0 タイマ C 後続エッジキャンセルモード
104 b0 NCMA = 1 タイマ A 先行エッジキャンセルモード */
105 ATUCTRL.NCMR.BYTE = 0x01; /* タイマ A のノイズキャンセラを先行エッジキャンセルに設定 */
106
107 /* Configure TIOR2A
108 b15:14 Reserved
109 b13 NCKA5 = 0 NCNTA5 のカウントソースクロックにノイズキャンセラカウントクロックを選択
110 b12 NCKA4 = 0 NCNTA4 のカウントソースクロックにノイズキャンセラカウントクロックを選択
111 b11 NCKA3 = 0 NCNTA3 のカウントソースクロックにノイズキャンセラカウントクロックを選択
112 b10 NCKA2 = 0 NCNTA2 のカウントソースクロックにノイズキャンセラカウントクロックを選択
113 b9 NCKA1 = 0 NCNTA1 のカウントソースクロックにノイズキャンセラカウントクロックを選択
114 b8 NCKA0 = 0 NCNTA0 のカウントソースクロックにノイズキャンセラカウントクロックを選択
115 b7:6 Reserved
116 b5 NCEA5 = 0 TIA05 入力のノイズキャンセル機能を無効
117 b4 NCEA4 = 0 TIA04 入力のノイズキャンセル機能を無効
118 b3 NCEA3 = 0 TIA03 入力のノイズキャンセル機能を無効
119 b2 NCEA2 = 0 TIA02 入力のノイズキャンセル機能を無効
120 b1 NCEA1 = 0 TIA01 入力のノイズキャンセル機能を無効
121 b0 NCEA0 = 1 TIA00 入力のノイズキャンセル機能を有効 */
122 ATUA.TIOR2A.WORD = 0x0001; /* TIA00 のノイズキャンセラを有効に設定
123     カウンタのクロックをクロックバス 0 に設定 */
124 /* Configure NCRA0
125 b7:0 NCCNTAn = H'12 ノイズキャンセルタイムの上限値設定 */
126 ATUA.NCRA0 = 0x12; /* TIA00 のノイズキャンセルカウンタの上限を 18 に設定 */
127
128 /* Configure PSCR0
129 b15:10 Reserved
130 b9:0 PSCn = H'1 プリスケーラの分周比を 2(設定値は-1)に設定 */
131 ATUCTRL.PSCR0.WORD = 0x0001; /* プリスケーラの分周比を 2 に設定 */
132
133 /* Configure ATUENR
134 b15:10 Reserved
135 b9 TJE = 0 タイマ J のカウント動作を停止
136 b8 THE = 0 タイマ H のカウント動作を停止
137 b7 TGE = 0 タイマ G のカウント動作を停止
138 b6 TFE = 0 タイマ F のカウント動作を停止
139 b5 TEE = 0 タイマ E のカウント動作を停止
140 b4 TDE = 0 タイマ D のカウント動作を停止
141 b3 TCE = 0 タイマ C のカウント動作を停止
142 b2 TBE = 0 タイマ B のカウント動作を停止
143 b1 TAE = 1 タイマ A のカウント動作を許可
144 b0 PSCE = 1 プリスケーラのカウント動作を許可 */

```

```
145    ATUCTRL.ATUENR.WORD = 0x0003; /* タイマAおよびプリスケアラのカウントをスタート */
146
147    while(1);
148 } /* End of function main() */
```

2.5.6 SH7253 グループでご使用の場合の変更点

本文 2.5 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.5.5 サンプルプログラムリストの 75 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.6 動作例 5 イベント周期測定[タイマ C]

2.6.1 概要

図 2.9に示すように、イベントの周期を測定し結果を RAM に格納します。外部からのイベントトリガを 2 回受けて、その間のカウンタ値を記録することでイベント間の周期を測定します。

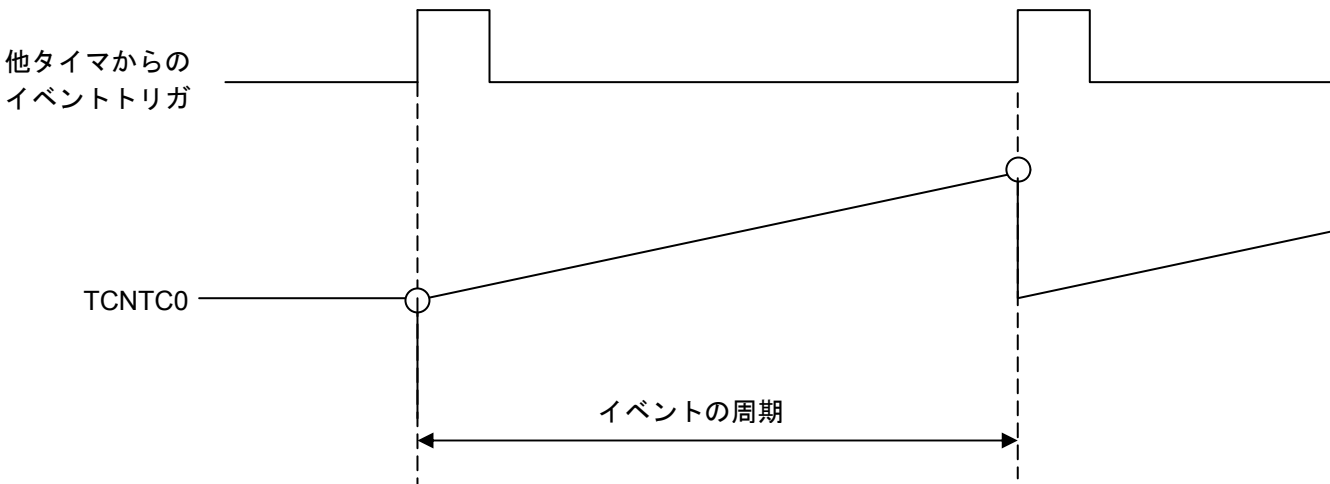


図 2.9 イベント周期測定タイミング

2.6.2 使用機能説明

表 1 に使用端子及び関連レジスタの機能割り付けを示します。

表 2.5 ATU-III機能割り付け

使用端子		機能
端子	TCLKA	測定するパルスはこの端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	CBCNT	クロックバス 4,5 に供給する外部入力信号の設定をします。
タイマ C レジスタ	TIORC1	タイマ C の I/O コントロールを設定します。
	TIERC1	TIC1 のインプットキャプチャ割り込みを許可/禁止を設定します
	TSRC0	インプットキャプチャが発生した場合にフラグがセットされます。
	GRC10	コンペアマッチの値を設定します。
	TSTRC	タイマ C のカウント動作を許可/停止します。
	TCNTC0	タイマ C の周期計算用のカウンタです。
	TCRC0	タイマ C の動作を設定します。
PFC レジスタ	PDCR2 PDCR1	端子機能を設定します。
INTC レジスタ	IPR08	IMIC10 のインプットキャプチャ割り込みの優先度を設定します。

2.6.3 動作説明

図 2.10に動作原理を示します。SH72546R のハードウェアおよびソフトウェア処理によってイベントの周期測定を行います。TIOC10 は外部からの入力で、立ち上がり/立下りは任意に行われます。

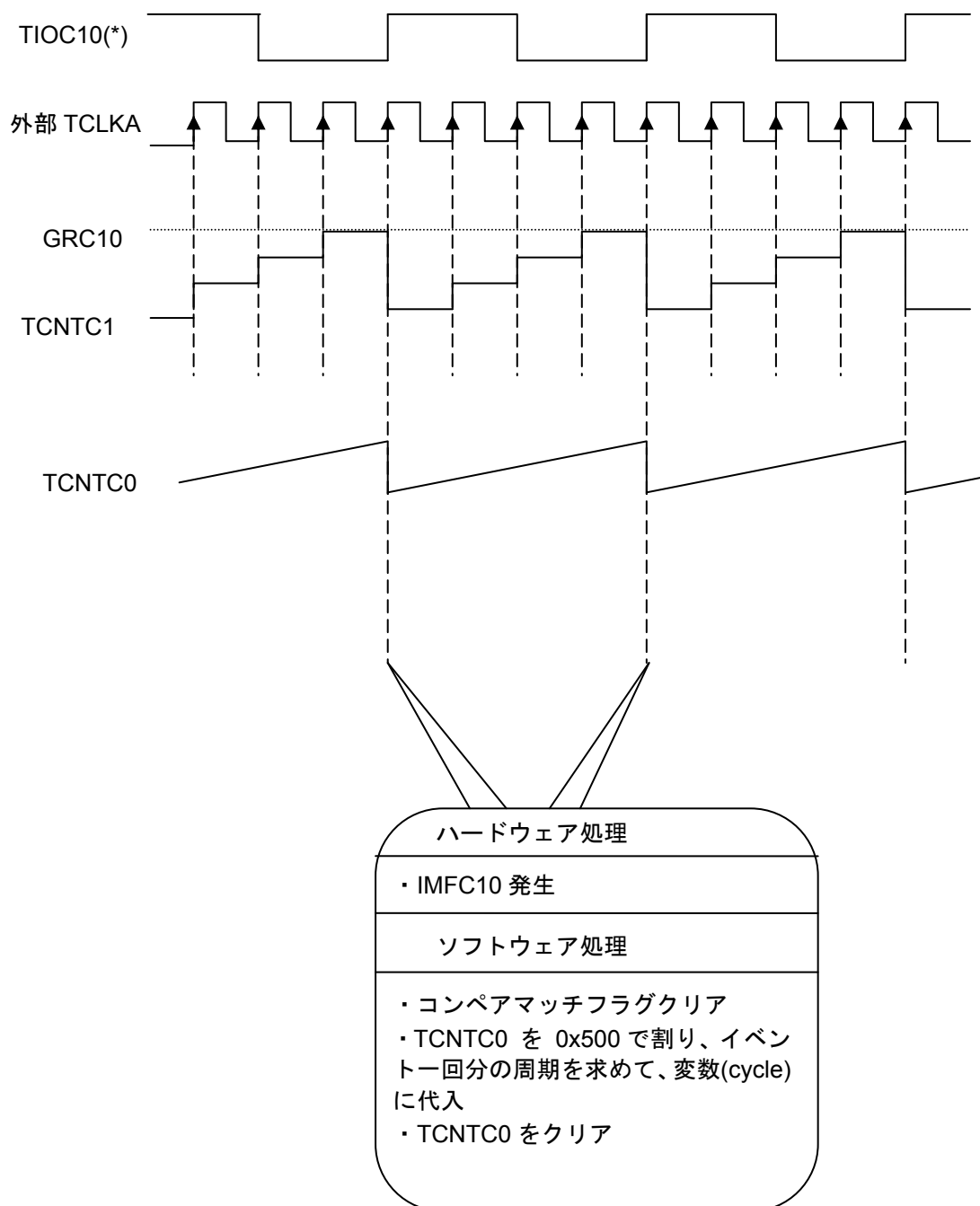


図 2.10 イベント周期計測動作原理

2.6.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
イベント周期測定	IMIC10	ICFA0 によって起動し、ICRA0 の値によってパルスの周期を測定します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
cycle	イベントの周期に相当するタイマ値を設定します。 イベントの周期は次の式で求められます。 イベントの周期(ns)=タイマ値×Pφ周期(40MHz 動作時 25ns)×10 (プリスケアラの分周比)	unsigned long	パルスの周期測定

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール
PDCR2	PD12 端子を TCLKA 機能(クロックバス 4 用外部入力)に設定します。	0x0100	メインルーチン
PDCR1	PD4 を TIOC10 入出力に設定します。	0x0100	
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIタイマ C 及びプリスケアラのカウント動作を設定します。	0x0009	
TIORC1	汎用レジスタ 10 のコンペアマッチ及び、コンペアマッチ発生時の動作を設定します。	0x0003	
TIERC1	TIOC10 に対応するコンペアマッチ割り込みを許可します。	0x01	
GRC10	TCNTC1 に対応するコンペアマッチの値を設定します。	0x00005000	
TCRC1	サブブロック 1 をクロックバス 0 で動作させます。	0x0C	
TSTRC	タイマ C のサブブロック 0、1 のカウントの動作を許可します。	0x03	
IPR08	TIOC00,01 の割り込み要求の優先度を設定します。	0xF000	イベント周期測定
TSRC1	タイマ C のコンペアマッチフラグをクリアします。	IMFC10 = 0	
TCNTC0	タイマカウンタをクリアします。	0x00000000	

2.6.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Periodic measurements of the event.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39 /*****
40 Includes <System Includes> , "Project Includes"
41 *****/
42 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
43 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
44
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  volatile unsigned long cycle;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  cycle = 0;
62
63  /* ポート設定 */
64  /* Configure PDCR2
65  b15:12 Reserved
66  b11:10 PD13MD = 00 PD13 入出力 (ポート)
67  b9:8 PD12MD = 01 PD12 TCLKA 入力 (ATU-III)
68  b7:6 PD11MD = 00 PD11 入出力 (ポート)
69  b5:4 PD10MD = 00 PD10 入出力 (ポート)
70  b3:2 PD9MD = 00 PD9 入出力 (ポート)
71  b1:0 PD8MD = 00 PD8 入出力 (ポート) */
72  PORTD.CR2.WORD = 0x0100; /* PD12 端子を TCLKA 機能(クロックバス 4 用外部入力)に設定 */
73
74  /* Configure PDCR1
75  b15:14 PD7MD = 00 PD7 入出力 (ポート)
76  b13:12 PD6MD = 00 PD6 入出力 (ポート)
77  b11:10 PD5MD = 00 PD5 入出力 (ポート)
78  b9:8 PD4MD = 01 PD4 TI0C10 入出力 (ATU-III)
79  b7:6 PD3MD = 00 PD3 入出力 (ポート)
80  b5:4 PD2MD = 00 PD2 入出力 (ポート)
81  b3:2 PD1MD = 00 PD1 入出力 (ポート)
82  b1:0 PD0MD = 00 PD0 入出力 (ポート) */
83  PORTD.CR1.WORD = 0x0100; /* PD4 端子を TI0C10 機能(コンペアマッチ入出力)に設定 */
84
85  /* ATU-III の設定 */
86  /* Configure TCRC
87  b7 FCMCn3 = 0 汎用レジスタ GRCn3 での強制コンペアマッチ禁止
88  b6 FCMCn2 = 0 汎用レジスタ GRCn2 での強制コンペアマッチ禁止
89  b5 FCMCn1 = 0 汎用レジスタ GRCn1 での強制コンペアマッチ禁止
90  b4 FCMCn0 = 0 汎用レジスタ GRCn0 での強制コンペアマッチ禁止
91  b3 PWNn0 = 1 PWM モードで動作
92  b2:0 CKSELCn = 100 クロックバス 4 でカウント */
93  ATUC.SUBBLOCK[1].TCRC.BYTE = 0x0C; /* サブブロック 1 をクロックバス 4(外部クロック入力
94  A)で動作 */

```

```

95  /* Configure TIERC
96  b7:5 Reserved
97  b4 OVEFn = 0 OVFCn による割り込み要求を禁止
98  b3 IMECn3 = 0 IMFCn3 による割り込み要求を禁止
99  b2 IMECn2 = 0 IMFCn2 による割り込み要求を禁止
100 b1 IMECn1 = 0 IMFCn1 による割り込み要求を禁止
101 b0 IMECn0 = 1 IMFCn0 による割り込み要求を許可 */
102 ATUC.SUBBLOCK[1].TIERC.BYTE = 0x01; /* TI0C10 に対応するコンペアマッチ割り込みを許可 */
103
104 /* Configure TIORC
105 b11 Reserved
106 b10:8 IOFn2 = 000 コンペアマッチ禁止
107 b7 Reserved
108 b6:4 IOFn1 = 000 コンペアマッチ禁止
109 b3 Reserved
110 b2:0 IOFn0 = 011 コンペアマッチでトリプル出力 */
111 ATUC.SUBBLOCK[1].TIORC.WORD = 0x003; /* 汎用レジスタ 10 のコンペアマッチでトリプル出力 */
112
113 /* Configure GRC
114 b31:0 汎用レジスタデータ */
115 ATUC.SUBBLOCK[1].GRC[0] = 0x00000500; /* コンペアマッチの値を 0x500 に設定 */
116
117 /* Configure TSTRC
118 b7:5 Reserved
119 b4 STRC4 = 0 TCNTC4 のカウント動作を停止
120 b3 STRC3 = 0 TCNTC3 のカウント動作を停止
121 b2 STRC2 = 0 TCNTC2 のカウント動作を停止
122 b1 STRC1 = 1 TCNTC1 のカウント動作を許可
123 b0 STRC0 = 1 TCNTC0 のカウント動作を許可 */
124 ATUC.TSTRC.BYTE = 0x03; /* タイマ C サブブロック 0,1 のカウントをスタート */
125
126 /* Configure CBCNT
127 b7:6 Reserved
128 b5:4 CB4EG = 01 クロックバス 4:外部クロックの立ち上がりエッジ選択
129 b3 Reserved
130 b2 CB5SEL = 0 クロックバス 5:外部クロック入力 B (TCLKB)
131 b1:0 CB5EG = 00 クロックバス 5:外部クロックのエッジ検出をしない */
132 ATUCTRL.CBCNT.BYTE = 0x10; /* クロックバス 4 は外部入力の立ち上がりエッジを元にカウント */
133
134 /* Configure PSCR0
135 b15:10 Reserved
136 b9:0 PSCn = H'9 プリスケアラの分周比を 10(設定値は-1)に設定 */
137 ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケアラの分周比を 10 に設定 */
138
139 /* 割り込み設定 */
140 /* Configure IPR08
141 b15:12 ATU-C1(IMIC10-13) = H'F 対応する割り込みの優先順位を 15 に設定
142 b11:8 ATU-C1(OVIC1) = H'0 対応する割り込みの優先順位を 0 に設定
143 b7:4 ATU-C2(IMIC20-23) = H'0 対応する割り込みの優先順位を 0 に設定
144 b3:0 ATU-C2(OVIC2) = H'0 対応する割り込みの優先順位を 0 に設定 */

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

145  INTC.IPR08.WORD = 0xF000; /* TI0C00,01 の割り込み優先順位を 15 に設定 */
146
147  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
148
149  /* Configure ATUENR
150  b15:10 Reserved
151  b9 TJE = 0 タイマ J のカウント動作を停止
152  b8 THE = 0 タイマ H のカウント動作を停止
153  b7 TGE = 0 タイマ G のカウント動作を停止
154  b6 TFE = 0 タイマ F のカウント動作を停止
155  b5 TEE = 0 タイマ E のカウント動作を停止
156  b4 TDE = 0 タイマ D のカウント動作を停止
157  b3 TCE = 1 タイマ C のカウント動作を許可
158  b2 TBE = 0 タイマ B のカウント動作を停止
159  b1 TAE = 0 タイマ A のカウント動作を停止
160  b0 PSCE = 1 プリスケーラのカウント動作を許可 */
161  ATUCTRL.ATUENR.WORD = 0x0009; /* タイマ C およびプリスケーラのカウントをスタート */
162
163  while(1); /* 無限ループに入り、待機 */
164  } /* End of function main() */
165
166  /*****
167  * Function Name : IMIC10
168  * Description    : コンペアマッチ割り込み
169  * Arguments      : none
170  * Return Value   : none
171  *****/
172  void IMIC10(void)
173  {
174  /* Configure TSRC
175  b7:5 Reserved
176  b4 OVFCn &= 1 オーバフロー発生
177  b3 IMFCn3 &= 1 インพุットキャプチャ、コンペアマッチフラグ維持
178  b2 IMFCn2 &= 1 インพุットキャプチャ、コンペアマッチフラグ維持
179  b1 IMFCn1 &= 1 インพุットキャプチャ、コンペアマッチフラグ維持
180  b0 IMFCn0 &= 0 インพุットキャプチャ、コンペアマッチフラグをクリア */
181  ATUC.SUBBLOCK[1].TSRC.BYTE &= 0xFE; /* コンペアマッチフラグをクリア */
182
183  /* 0x500 で割り、イベント一回の周期を求める */
184  cycle = (ATUC.SUBBLOCK[0].TCNTC / ATUC.SUBBLOCK[1].GRC[0]);
185
186  /* Configure TCNTC
187  b31:0 タイマカウンタ */
188  ATUC.SUBBLOCK[0].TCNTC ^= ATUC.SUBBLOCK[0].TCNTC;
189  } /* End of function IMIC10() */

```

2.6.6 SH7253 グループでご使用の場合の変更点

本文 2.6 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください。
- (2)TCLKA 端子が、SH7253 グループでは、PG10 端子 (SH7254R グループでは、PD12 端子) に割り当てられています。また、TIOC10 端子が、SH7253 グループでは、PA4 端子 (SH7254R グループでは、PD4 端子) に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.6.5 サンプルプログラムリストの 72 行目を

PORTG.CR3A.WORD=0x0100 に変更

SH7254R グループ : PDCR2 を 0x0100 に設定 (変更前)

SH7253 グループ : PGCR3A を 0x0100 に設定 (変更後)

2.6.5 サンプルプログラムリストの 83 行目を

PORTA.CR2A.WORD=0x0001 に変更

SH7254R グループ : PDCR1 を 0x0100 に設定 (変更前)

SH7253 グループ : PACR2A を 0x0001 に設定 (変更後)

2.7 動作例 6 パルス出力(トグル出力、タイマカウントクリア)[タイマ C]

2.7.1 概要

図 2.11に示すように、一定の周期でパルスをトグル出力します。本機能を実現する為に、タイマ動作モードを PWM モードに設定します。

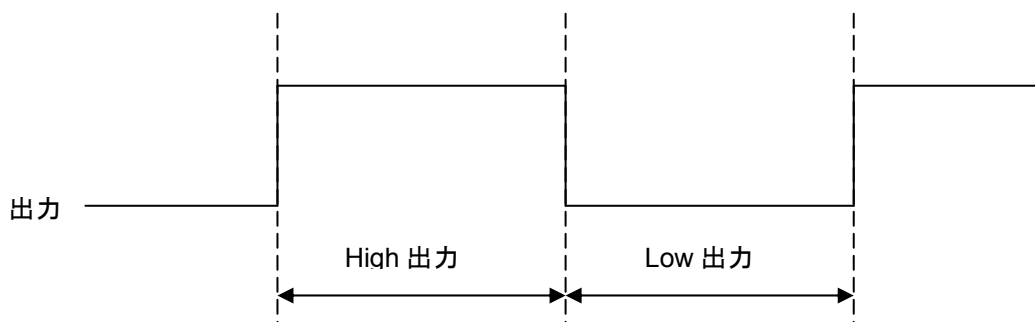


図 2.11 トグル出力

2.7.2 使用機能説明

表 2.6に使用端子及び関連レジスタの機能割り付けを示します。

表 2.6 ATU-III機能割り付け

使用端子		機能
端子	TIOC00	パルスを出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ C レジスタ	TIORC0	タイマ C の機能を設定します。
	TCRC0	タイマ C の動作モード、クロックバスを設定します。
	GRC00	周期を設定します。
	TSTRC	タイマ C 各サブブロックのカウンタの動作を設定します。
PFC レジスタ	PDCR1	端子機能を設定します。

2.7.3 動作説明

図 2.12に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってパルス出力を行います。PWM モードでは、TCNTC0 と GCR00 とのコンペアマッチ発生時に TCNTC0 の値は 0 にクリアされ、TCNTC0 は再度 0 からカウンタ動作を行います。

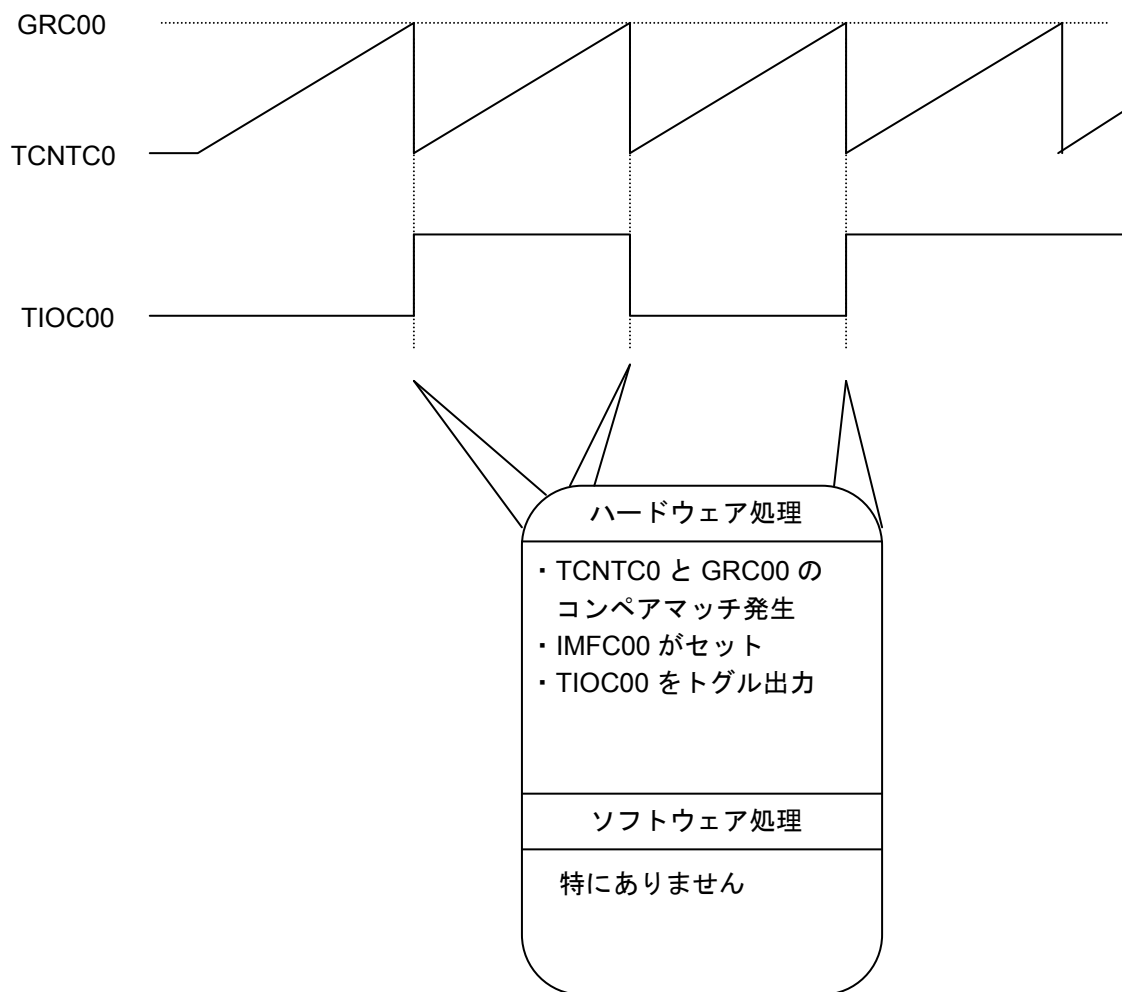


図 2.12 パルス出力動作原理

2.7.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。

- 使用変数の説明

本タスクでは変数は使用していません。

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール
PDCR1	ポート D0 端子を入力端子 TIOC00 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	タイマ C 及びプリスケアラのカウンタ動作を設定します。	0x0009	
TIORC0	汎用レジスタ 00 のコンペアマッチ及び、コンペアマッチ発生時の動作を設定します。	0x0003	
TCRC0	サブブロック C0 のクロック選択、動作モードの設定、強制コンペアマッチの設定を行います。	0x08	
GRC00	TCNTC0 に対応するコンペアマッチの値を設定する。	0x00005000	
TSTRC	サブブロック 0、1 のカウンタの動作許可をします。	0x01	

2.7.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Pulse output (toggle output, timer counter clear).
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments    : none
55  * Return Value  : none
56  *****/
57  void main(void)
58  {
59  /* ポートの設定 */
60  /* Configure PDCR1
61  b15:14 PD7MD = 00 PD7 入出力 (ポート)
62  b13:12 PD6MD = 00 PD6 入出力 (ポート)
63  b11:10 PD5MD = 00 PD5 入出力 (ポート)
64  b9:8 PD4MD = 01 PD4 入出力 (ポート)
65  b7:6 PD3MD = 00 PD3 入出力 (ポート)
66  b5:4 PD2MD = 00 PD2 入出力 (ポート)
67  b3:2 PD1MD = 00 PD1 入出力 (ポート)
68  b1:0 PD0MD = 01 PD0 TI0C00 入出力 (ATU-III) */
69  PORTD.CR1.WORD = 0x0001; /* PD0 を TI0C00 の入出力に設定(出力波形) */
70
71  /* ATU-IIIの設定 */
72  /* Configure TSTRC
73  b7:5 Reserved
74  b4 STRC4 = 0 TCNTC4 のカウント動作を停止
75  b3 STRC3 = 0 TCNTC3 のカウント動作を停止
76  b2 STRC2 = 0 TCNTC2 のカウント動作を停止
77  b1 STRC1 = 0 TCNTC1 のカウント動作を停止
78  b0 STRC0 = 1 TCNTC0 のカウント動作を許可 */
79  ATUC.TSTRC.BYTE = 0x01; /* タイマ C サブブロック 0 のカウントをスタート */
80
81  /* Configure TCRC
82  b7 FCMCn3 = 0 汎用レジスタ GRCn3 での強制コンペアマッチ禁止
83  b6 FCMCn2 = 0 汎用レジスタ GRCn2 での強制コンペアマッチ禁止
84  b5 FCMCn1 = 0 汎用レジスタ GRCn1 での強制コンペアマッチ禁止
85  b4 FCMCn0 = 0 汎用レジスタ GRCn0 での強制コンペアマッチ禁止
86  b3 PWNn0 = 1 PWM モードで動作
87  b2:0 CKSELCn = 000 クロックバス 0 でカウント */
88  ATUC.SUBBLOCK[0].TCRC.BYTE = 0x08; /* サブブロック 1 をクロックバス 0(外部クロック入力
      A)で動作 */
89
90  /* Configure TI0RC
91  b15 Reserved
92  b14:12 IOCN3 = 000 汎用レジスタ 03 コンペアマッチ禁止
93  b11 Reserved

```

アドバンスタイマユニット-III (ATU-III) を使用したタイマ動作例

```
94    b10:8 IOCn2 = 000 汎用レジスタ 02 コンペアマッチ禁止
95    b7 Reserved
96    b6:4 IOCn1 = 000 汎用レジスタ 01 コンペアマッチ禁止
97    b3 Reserved
98    b2:0 IOCn0 = 011 汎用レジスタ 00 コンペアマッチでトグル出力 */
99    ATUC.SUBBLOCK[0].TIORC.WORD = 0x0003; /* 汎用レジスタ 00 のコンペアマッチでトグル出力 */
100
101    /* Configure GRC
102    b31:0 汎用レジスタデータ(コンペアマッチに使用) */
103    ATUC.SUBBLOCK[0].GRC[0] = 0x00005000; /* コンペアマッチの値を 0x50 に設定 */
104
105    /* Configure PSCR0
106    b15:10 Reserved
107    b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
108    ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
109
110    /* Configure ATUENR
111    b15:10 Reserved
112    b9 TJE = 0 タイマ J のカウント動作を停止
113    b8 THE = 0 タイマ H のカウント動作を停止
114    b7 TGE = 0 タイマ G のカウント動作を停止
115    b6 TFE = 0 タイマ F のカウント動作を停止
116    b5 TEE = 0 タイマ E のカウント動作を停止
117    b4 TDE = 0 タイマ D のカウント動作を停止
118    b3 TCE = 1 タイマ C のカウント動作を許可
119    b2 TBE = 0 タイマ B のカウント動作を停止
120    b1 TAE = 0 タイマ A のカウント動作を停止
121    b0 PSCE = 1 プリスケーラのカウント動作を許可 */
122    ATUCTRL.ATUENR.WORD = 0x0009; /* タイマ C およびプリスケーラのカウントを動作 */
123
124    while(1);
125    } /* End of function main() */
```

2.7.6 SH7253 グループでご使用の場合の変更点

本文 2.7 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIOC00 端子が、SH7253 グループでは、PA0 端子（SH7254R グループでは、PD0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.7.5 サンプルプログラムリストの 69 行目を

PORTA.CR1A.WORD=0x0001 に変更

SH7254R グループ : PDCR を 0x0001 に設定 (変更前)

SH7253 グループ : PACR1A を 0x0001 に設定 (変更後)

2.8 動作例 7 パルス出力(トグル出力、コンペア値加算)[タイマ C]

2.8.1 概要

図 2.13に示すように、一定の周期でパルスをトグル出力します。本動作例では HIGH と Low の切り替えタイミングをコンペアマッチ値の加算で更新することにより、パルス出力を実現します。

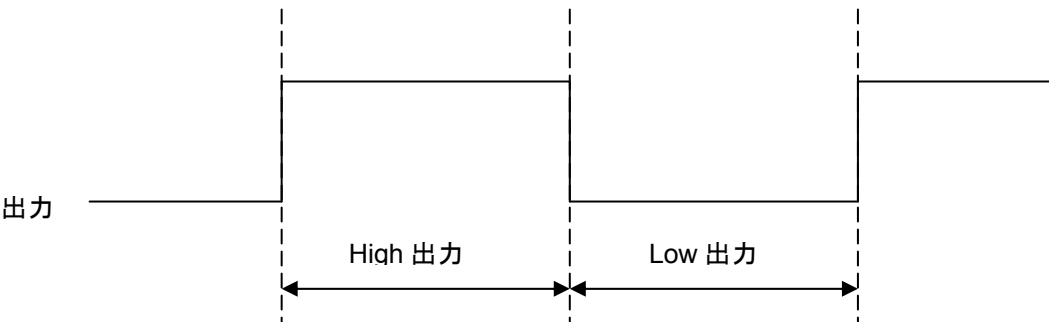


図 2.13 トグル出力

2.8.2 使用機能説明

表 2.7に使用端子及び関連レジスタの機能割り付けを示します。

表 2.7 ATU-III機能割り付け

使用端子		機能
端子	TIOC00	パルスを出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ C レジスタ	TIORC0	タイマ C の機能を設定します。
	TIERC	インプットキャプチャ/コンペアマッチ割り込みの許可/禁止を設定します
	TSRC	インプットキャプチャ/コンペアマッチ発生時にフラグがセットされます。
	GRC	周期を設定します。
	TSTRC	タイマ C 各サブブロックのカウンタの動作を設定します。
PFC レジスタ	PDCR1	端子機能を設定します。
INTC レジスタ	IPR07	IMIC00 のコンペアマッチ割り込みの優先度を設定します。

2.8.3 動作説明

図 2.14に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってパルス出力を行います。タイマカウンタ TCNTC0 とコンペアマッチ用レジスタ GRC00 とのコンペアマッチが発生すると、GRC00 は自身を加算します。

本動作例では GRC00 の値は常に一定に増加するので、一定の High/Low 幅を持つパルスが出力されます。

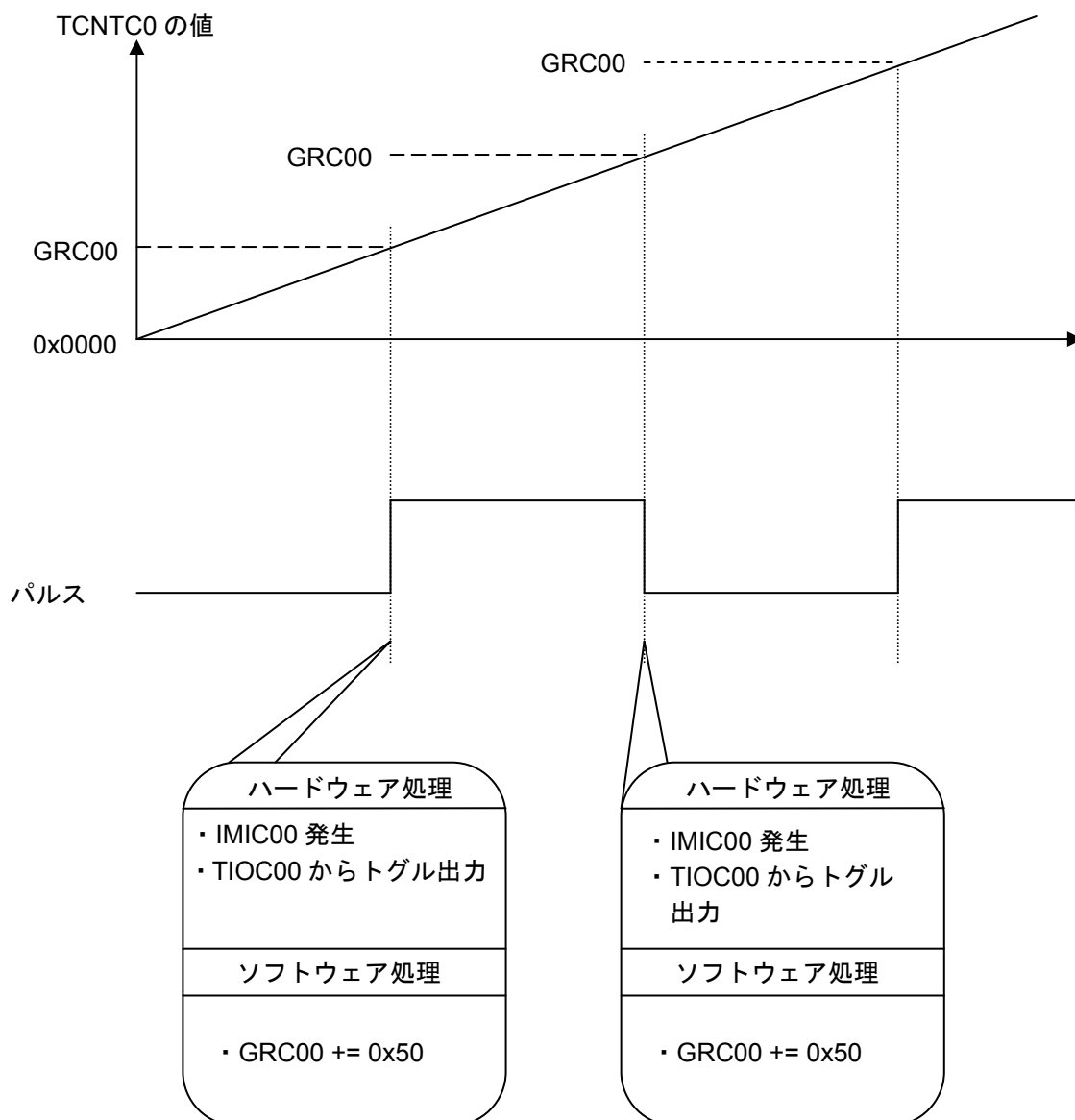


図 2.14 パルス出力動作原理

2.8.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
再コンペアマッチルーチン	IMIC00	ICFC00 によって起動し、TSRC をクリアし、汎用レジスタ 00 の値に加算を行う事によってコンペアマッチを再び発生可能にします。

● 使用変数の説明

本タスクでは変数は使用していません。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール
PDCR1	ポート D0 端子を入出力端子 TIOC00 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	タイマ C 及びプリスケアラのカウンタ動作を設定します。	0x0009	
TIORC0	汎用レジスタ 00 のコンペアマッチ及び、コンペアマッチ発生時の動作を設定します。	0x0003	
TIERC0	TIOC10 に対応するコンペアマッチ割り込みを許可します。	0x01	
GRC00	TCNTC1 に対応するコンペアマッチの値を設定する。	0x00005000	
TSTRC0	サブブロック 0 のカウンタの動作許可をします。	0x01	
IPR07	TCNTC0 のコンペアマッチ割り込み要求の優先度を設定します。	0x00FF	

2.8.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Measurement of Pulse period.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39 /*****
40 Includes <System Includes> , "Project Includes"
41 *****/
42 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
43 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
44
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments    : none
55  * Return Value  : none
56  *****/
57
58  void main(void)
59  {
60  /* ポートの設定 */
61  /* Configure PDCR1
62  b15:14 PD7MD = 00 PD7 入出力 (ポート)
63  b13:12 PD6MD = 00 PD6 入出力 (ポート)
64  b11:10 PD5MD = 00 PD5 入出力 (ポート)
65  b9:8 PD4MD = 01 PD4 入出力 (ポート)
66  b7:6 PD3MD = 00 PD3 入出力 (ポート)
67  b5:4 PD2MD = 00 PD2 入出力 (ポート)
68  b3:2 PD1MD = 00 PD1 入出力 (ポート)
69  b1:0 PD0MD = 01 PD0 TIOC00 入出力 (ATU-III) */
70  PORTD.CR1.WORD = 0x0001; /* PD0 を TIOC00 の入出力に設定(出力波形) */
71
72  /* ATU-IIIの設定 */
73  /* Configure TSTRC
74  b7:5 Reserved
75  b4 STRC4 = 0 TCNTC4 のカウント動作を停止
76  b3 STRC3 = 0 TCNTC3 のカウント動作を停止
77  b2 STRC2 = 0 TCNTC2 のカウント動作を停止
78  b1 STRC1 = 0 TCNTC1 のカウント動作を停止
79  b0 STRC0 = 1 TCNTC0 のカウント動作を許可 */
80  ATUC.TSTRC.BYTE = 0x01; /* タイマ C サブブロック 0 のカウントをスタート */
81
82  /* Configure TIERC
83  b7:5 Reserved
84  b4 OVEFn = 0 OVFCn による割り込み要求を禁止
85  b3 IMECn3 = 0 IMFCn3 による割り込み要求を禁止
86  b2 IMECn2 = 0 IMFCn2 による割り込み要求を禁止
87  b1 IMECn1 = 0 IMFCn1 による割り込み要求を禁止
88  b0 IMECn0 = 1 IMFCn0 による割り込み要求を許可 */
89  ATUC.SUBBLOCK[0].TIERC.BYTE = 0x01; /* コンペアマッチフラグ 00 によるコンペアマッチ割り
    込みを許可 */
90
91  /* Configure TIORC
92  b15 Reserved
93  b14:12 IOCn3 = 000 汎用レジスタ 03 コンペアマッチ禁止

```



```

94     b11 Reserved
95     b10:8 IOCN2 = 000 汎用レジスタ 02 コンペアマッチ禁止
96     b7 Reserved
97     b6:4 IOCN1 = 000 汎用レジスタ 01 コンペアマッチ禁止
98     b3 Reserved
99     b2:0 IOCN0 = 011 汎用レジスタ 00 コンペアマッチでトグル出力 */
100    ATUC.SUBBLOCK[0].TIORC.WORD = 0x0003; /* 汎用レジスタ 00 のコンペアマッチでトグル出力 */
101
102    /* Configure GRC
103    b31:0 汎用レジスタデータ(コンペアマッチに使用) */
104    ATUC.SUBBLOCK[0].GRC[0] = 0x00005000; /* コンペアマッチの値を H'50 に設定 */
105
106    /* Configure PSCR0
107    b15:10 Reserved
108    b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
109    ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
110
111    /* Configure IPR07
112    b15:12 ATU-B(CMIB0,1) = H'0 対応する割り込みの優先順位を 0 に設定
113    b11:8 ATU-B(CMIB6,ICIB0) = H'0 対応する割り込みの優先順位を 0 に設定
114    b7:4 ATU-C0(IMIC00-03) = H'F 対応する割り込みの優先順位を 15 に設定
115    b3:0 ATU-C0(OVIC0) = H'F 対応する割り込みの優先順位を 15 に設定 */
116    INTC.IPR07.WORD = 0x00FF; /* IMIC00、OVIC0 の割り込み優先順位を 15 に設定 */
117
118    set_imask(0); /* 割り込みマスケベルを 0 に設定 */
119
120    /* Configure ATUENR
121    b15:10 Reserved
122    b9 TJE = 0 タイマ J のカウント動作を停止
123    b8 THE = 0 タイマ H のカウント動作を停止
124    b7 TGE = 0 タイマ G のカウント動作を停止
125    b6 TFE = 0 タイマ F のカウント動作を停止
126    b5 TEE = 0 タイマ E のカウント動作を停止
127    b4 TDE = 0 タイマ D のカウント動作を停止
128    b3 TCE = 1 タイマ C のカウント動作を許可
129    b2 TBE = 0 タイマ B のカウント動作を停止
130    b1 TAE = 0 タイマ A のカウント動作を停止
131    b0 PSCE = 1 プリスケーラのカウント動作を許可 */
132    ATUCTRL.ATUENR.WORD = 0x0009; /* タイマ C およびプリスケーラのカウントを動作 */
133
134    while(1);
135    } /* End of function main() */
136
137    /*****
138    * Function Name : IMIC00
139    * Description   : コンペアマッチ割り込み
140    * Arguments    : none
141    * Return Value  : none
142    *****/
143    void IMIC00(void)

```

```
144  {
145  /* Configure TSRC
146  b7:5 Reserved
147  b4 OVFCn &= 1 オーバフロー発生
148  b3 IMFCn3 &= 1 インพุットキャプチャ、コンペアマッチフラグを維持
149  b2 IMFCn2 &= 1 インพุットキャプチャ、コンペアマッチフラグを維持
150  b1 IMFCn1 &= 1 インพุットキャプチャ、コンペアマッチフラグを維持
151  b0 IMFCn0 &= 0 インพุットキャプチャ、コンペアマッチフラグをクリア */
152  ATUC.SUBBLOCK[0].TSRC.BYTE &= 0xFE; /* コンペアマッチフラグをクリア */
153
154  /* Configure GRC
155  b31:0 汎用レジスタデータ(コンペアマッチに使用) */
156  ATUC.SUBBLOCK[0].GRC[0] += 0x5000; /* 汎用レジスタのコンペアマッチの値を 0x50 増加 */
157  } /* End of function IMIC00() */
```

2.8.6 SH7253 グループでご使用の場合の変更点

本文 2.8 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIOC00 端子が、SH7253 グループでは、PA0 端子（SH7254R グループでは、PD0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.8.5 サンプルプログラムリストの 70 行目を

PORTA.CR1A.WORD=0x0001 に変更

SH7254R グループ : PDCR を 0x0001 に設定 (変更前)

SH7253 グループ : PACR1A を 0x0001 に設定 (変更後)

2.9 動作例 8 パルス出力(High/Low 切替出力、コンペア値加算)[タイマ C]

2.9.1 概要

図 2.15に示すように、一定の周期でパルスをトグル出力します。

本動作例ではポート出力の HIGH と Low の切り替えタイミングを、タイマのコンペアマッチ値加算で更新します。コンペアマッチ発生毎にポートの High/Low を切り替えることで、パルス出力を行います。

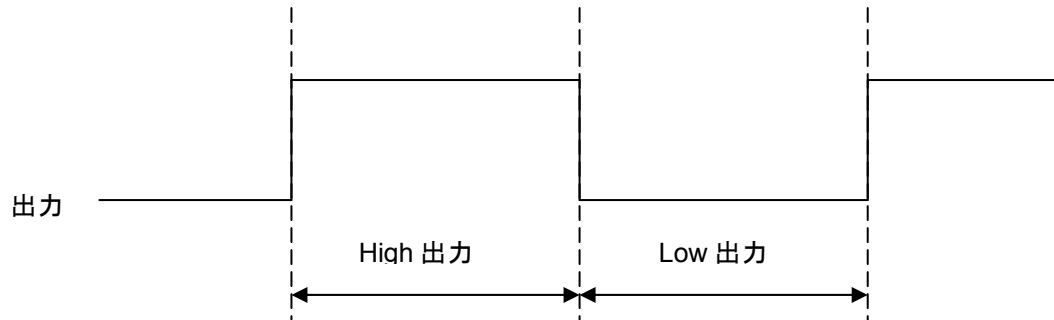


図 2.15 トグル出力

2.9.2 使用機能説明

表 2.8に使用端子及び関連レジスタの機能割り付けを示します。

表 2.8 ATU-III機能割り付け

使用端子		機能
端子	PE0	設定されたレベルのパルスを出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ C レジスタ	TIORC	タイマ C の機能を設定します。
	TIERC	インプットキャプチャ/コンペアマッチ割り込みの許可/禁止を設定します
	TSRC	インプットキャプチャ/コンペアマッチ発生時にフラグがセットされます。
	GRC00	周期を設定します。
	TSTRC	タイマ C 各サブブロックのカウンタの動作を設定します。
PFC レジスタ	PEIOR	PE0 端子の入出力方向を設定します。
	PEDR	PE0 の出力レベルを設定します。
INTC レジスタ	IPR07	IMIC00 のコンペアマッチ割り込みの優先度を設定します。

2.9.3 動作説明

図 2.16に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってパルス出力を行います。タイマカウンタ TCNTC0 とコンペアマッチ用レジスタ GRC00 とのコンペアマッチが発生すると、GRC00 は自身を加算します。

本動作例では GRC00 の値は常に一定に増加するので、一定の High/Low 幅を持つパルスが出力されます。

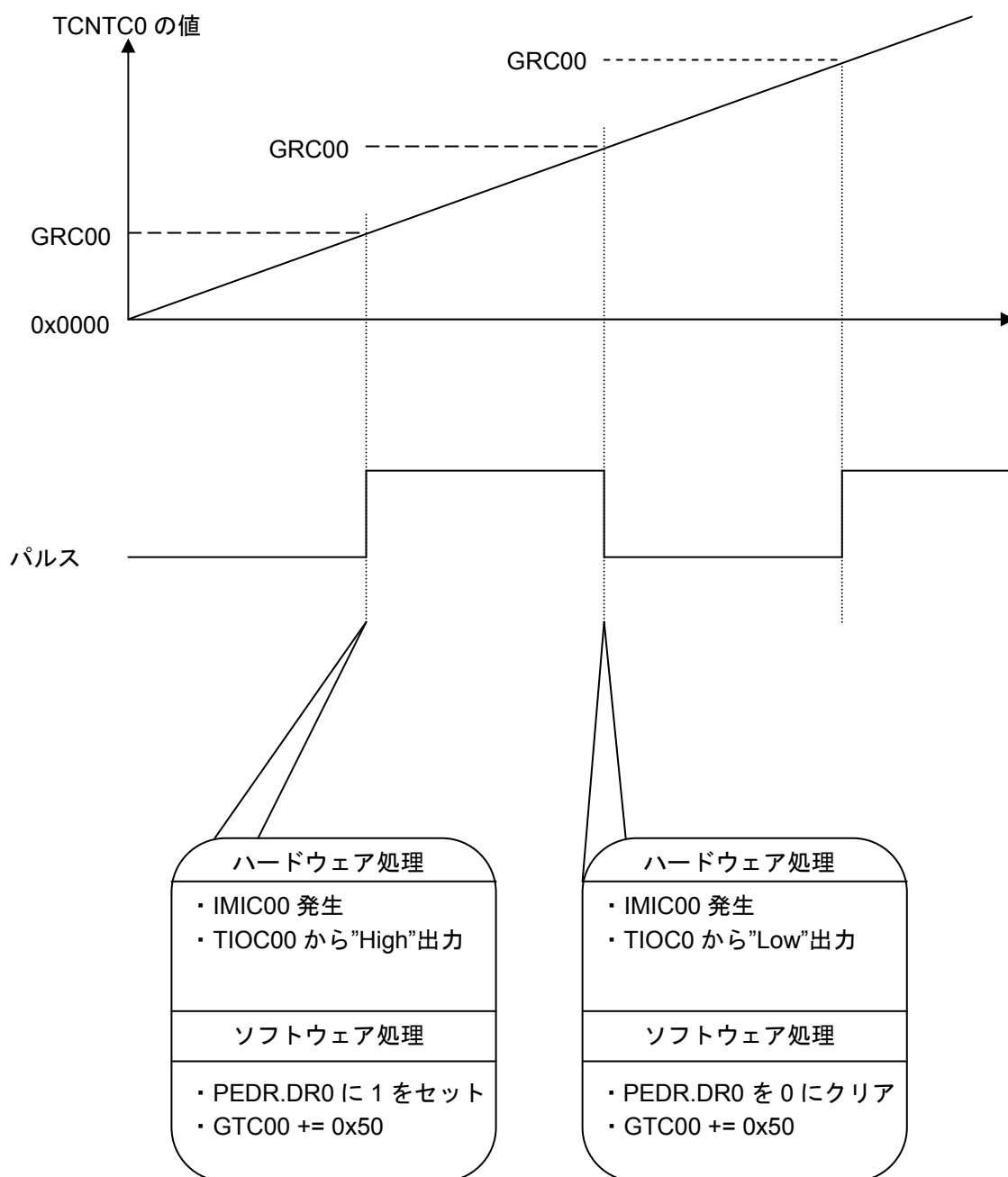


図 2.16 パルス出力動作原理

2.9.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
出力の切り替えルーチン	IMIC00	ICFC00 によって起動し、PE0 の出力レベルの切り替え、および再コンペアマッチへの処理を行います

- 使用変数の説明

本タスクでは変数は使用していません。

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール
PEIOR	ポート PE0 端子を汎用出力に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-III タイマ C 及びプリスケアラのカウント動作を設定します。	0x0009	
TIORC0	汎用レジスタ 00 のコンペアマッチ及び、コンペアマッチ発生時の動作を設定します。	0x0001	
TIERC0	TIOC10 に対応するコンペアマッチ割り込みを許可します。	0x01	
GRC00	TCNTC1 に対応するコンペアマッチの値を設定する。	0x00005000	
TSTRC	サブブロック 0、1 のカウントの動作の許可を設定します。	0x01	
IPR07	TCNTC0 のコンペアマッチ割り込み要求の優先度を設定します。	0x00FF	
PEDR	PE0 の出力初期値を Low に設定します。	0x00	

2.9.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Pulse output.(Switchable output High / Low, compare the added value)
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments    : none
55  * Return Value  : none
56  *****/
57
58  void main(void)
59  {
60  /* ポートの設定 */
61  /* Configure PEI0R
62  b15:14 Reserved
63  b13:0 PEnI0R 対応する端子の入出力設定 0:Input 1:Output */
64  PORTE.I0R.WORD = 0x0001; /* PE0 を出力に設定 */
65
66  /* Configure PEDR
67  b15:14 Reserved
68  b13:0 RENDR = H'001 対応する端子から "1" を出力 */
69  PORTE.DR.BIT.DR0 = 1; /* PE0 の初期値を High に設定 */
70
71  /* ATU-III の設定 */
72  /* Configure TSTRC
73  b7:5 Reserved
74  b4 STRC4 = 0 TCNTC4 のカウント動作を停止
75  b3 STRC3 = 0 TCNTC3 のカウント動作を停止
76  b2 STRC2 = 0 TCNTC2 のカウント動作を停止
77  b1 STRC1 = 0 TCNTC1 のカウント動作を停止
78  b0 STRC0 = 1 TCNTC0 のカウント動作を許可 */
79  ATUC.TSTRC.BYTE = 0x01; /* サブブロック 0 のカウントを動作 */
80
81  /* Configure TIERC
82  b7:5 Reserved
83  b4 OVECn = 0 OVFCn による割り込み要求を禁止
84  b3 IMECn3 = 0 IMFCn3 による割り込み要求を禁止
85  b2 IMECn2 = 0 IMFCn2 による割り込み要求を禁止
86  b1 IMECn1 = 0 IMFCn1 による割り込み要求を禁止
87  b0 IMECn0 = 1 IMFCn0 による割り込み要求を許可 */
88  ATUC.SUBBLOCK[0].TIERC.BYTE = 0x01; /* コンペアマッチフラグ 00 によるコンペアマッチ割り
89  込みを許可 */
90
91  /* Configure TI0RC
92  b15 Reserved
93  b14:12 IOCN3 = 000 コンペアマッチ禁止
94  b11 Reserved

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94    b10:8 IOcn2 = 000 コンペアマッチ禁止
95    b7 Reserved
96    b6:4 IOcn1 = 000 コンペアマッチ禁止
97    b3 Reserved
98    b2:0 IOcn0 = 001 コンペアマッチで 0 出力 */
99    ATUC.SUBBLOCK[0].TIORC.WORD = 0x0001; /* 汎用レジスタ 00 のインプットキャプチャを禁止 */
100
101    /* Configure GRC
102    b31:0 汎用レジスタデータ */
103    ATUC.SUBBLOCK[0].GRC[0] = 0x00005000; /* コンペアマッチの値を 0x5000 に設定 */
104
105    /* Configure PSCR0
106    b15:10 Reserved
107    b9:0 PSCn = H'9 プリスケアラの分周比を 10(設定値は-1)に設定 */
108    ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケアラの分週比を 10 に設定 */
109
110    /* Configure IPR07
111    b15:12 ATU-B(CMIB0,1) = H'0 対応する割り込みの優先順位を 15 に設定
112    b11:8 ATU-B(CMIB6,ICIB0) = H'0 対応する割り込みの優先順位を 0 に設定
113    b7:4 ATU-C0(IMIC00-03) = H'F 対応する割り込みの優先順位を 0 に設定
114    b3:0 ATU-C0(OVIC0) = H'F 対応する割り込みの優先順位を 0 に設定 */
115    INTC.IPR07.WORD = 0x00FF; /* IMIC00、OVIC0 の割り込み優先順位を 15 に設定 */
116
117    set_imask(0); /* 割り込みマスケレベルを 0 に設定 */
118
119    /* Configure ATUENR
120    b15:10 Reserved
121    b9 TJE = 0 タイマ J のカウント動作を停止
122    b8 THE = 0 タイマ H のカウント動作を停止
123    b7 TGE = 0 タイマ G のカウント動作を停止
124    b6 TFE = 0 タイマ F のカウント動作を停止
125    b5 TEE = 0 タイマ E のカウント動作を停止
126    b4 TDE = 0 タイマ D のカウント動作を停止
127    b3 TCE = 1 タイマ C のカウント動作を許可
128    b2 TBE = 0 タイマ B のカウント動作を停止
129    b1 TAE = 0 タイマ A のカウント動作を停止
130    b0 PSCE = 1 プリスケアラのカウント動作を許可 */
131    ATUCTRL.ATUENR.WORD = 0x0009; /* タイマ C およびプリスケアラのカウントを動作 */
132
133    while(1);
134    } /* End of function main() */
135
136    /*****
137    * Function Name : IMIC00
138    * Description   : コンペアマッチ割り込み
139    * Arguments     : none
140    * Return Value  : none
141    *****/
142    void IMIC00(void)
143    {

```

```
144  /* Configure TSRC
145  b7:5 Reserved
146  b4 OVFCn &= 1 オーバフローフラグを維持
147  b3 IMFCn3 &= 1 インพุットキャプチャ、コンペアマッチフラグを維持
148  b2 IMFCn2 &= 1 インพุットキャプチャ、コンペアマッチフラグを維持
149  b1 IMFCn1 &= 1 インพุットキャプチャ、コンペアマッチフラグを維持
150  b0 IMFCn0 &= 0 インพุットキャプチャ、コンペアマッチフラグをクリア */
151  ATUC.SUBBLOCK[0].TSRC.BYTE &= 0xFE; /* コンペアマッチフラグをクリア */
152
153  /* Configure GRC
154  b31:0 汎用レジスタデータ(コンペアマッチに使用) */
155  ATUC.SUBBLOCK[0].GRC[0] += 0x5000; /* 汎用レジスタのコンペアマッチの値を 0x50 増加 */
156
157  /* Configure PEDR
158  b15:14 Reserved
159  b13:0 RENDR 対応する端子から "1" を出力 (b0 = DR.BIT.DR0) */
160  if(PORTE.DR.BIT.DR0 == 0) /* 出力が Low ならば */
161  {
162  PORTE.DR.BIT.DR0 = 1; /* 出力を High に設定 */
163  }
164  else /* 出力を High にならば */
165  {
166  PORTE.DR.BIT.DR0 = 0; /* 出力を Low に設定 */
167  }
168  } /* End of function IMIC00() */
```

2.9.6 SH7253 グループでご使用の場合の変更点

本文 2.9 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.9.5 サンプルプログラムリストの 77 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

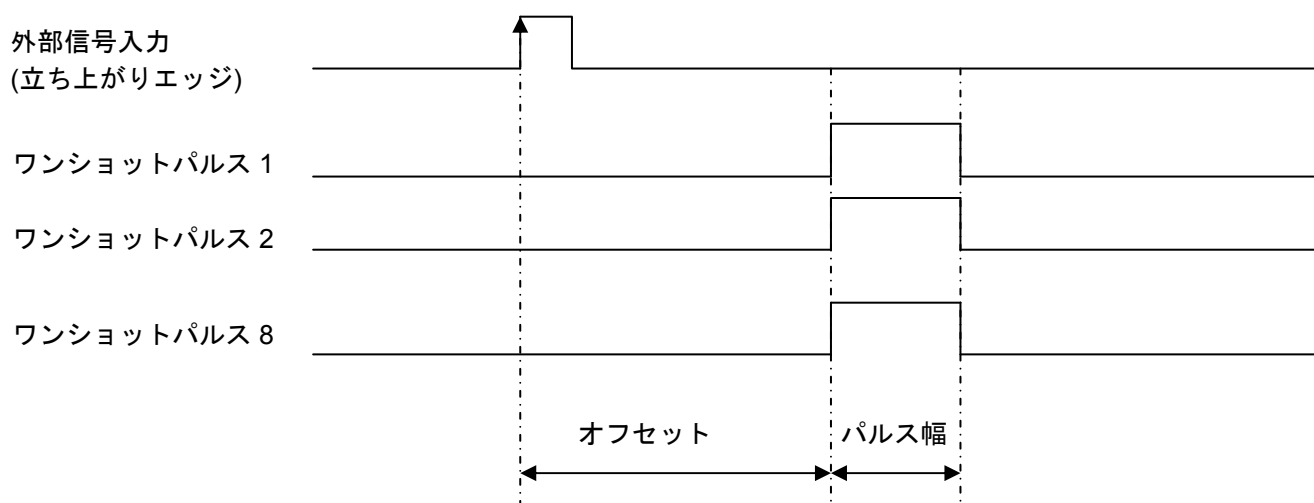
2.10 動作例 9 ワンショットパルス出力(オフセット付)[タイマ D]

2.10.1 概要

- 1) 図 2.17に示すように外部信号の立ち上がり同期してワンショットパルスを出力します。オフセット、パルス幅は内部クロックカウント値を設定します。
- 2) 本タスク例では、外部信号の立ち上がりからのオフセット及びパルス幅は以下に示す範囲で可変可能です。

$$4\mu\text{s} < \text{オフセット} < P\phi \text{ 周期} \times 65536$$

$$\text{*注: } 200\text{ns} \leq \text{パルス幅} < 13\text{ms}$$



*注：オフセットは外部信号の周期より小さく、
オフセット設定ルーチンの実行時間(約 $4\mu\text{s}$)より大きいこと。

図 2.17 ワンショットパルス出力

2.10.2 使用機能説明

表 2.9に使用端子及び関連レジスタの機能割り付けを示します。

表 2.9 ATU-III機能割り付け

使用端子		機能
端子	TIA00	開始用の外部信号を入力します。
	TOD00B～02B	ワンショットパルスを出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ A レジスタ	TIOR1A	キャプチャする外部入力のエッジの向きを設定します。
	TSRA0	外部入力のインプットキャプチャが発生した場合、フラグがセットされます。
	TIERA0	TIA00 のインプットキャプチャ割り込みを許可します。
タイマ D レジスタ	TIOR1D	OCRD に対応するコンペアマッチの許可、および発生時の動作の設定を行います。
	DCRD0	ダウンカウンタの開始、停止条件を設定します。
	TSRD0	コンペアマッチ及び、オーバー、アンダー両フローが発生した場合、フラグがセットされます。
	TSTRD	タイマ D のサブブロックのカウント動作を設定します。
	OCRD0	タイマ D のアウトプットコンペアレジスタです。
	DCNTD0	タイマ D のダウンカウンタです。
INTC レジスタ	IPR06	TIA00 のコンペアマッチ割り込みの優先度を設定します。
PFC レジスタ	PECR1	ポート E 端子の機能を設定します。
	PFCR1	ポート F 端子の機能を設定します。

2.10.3 動作説明

図 2.18に動作原理を示します。図に示すように SH72546R のハードウェア及びソフトウェアの処理によりワンショットパルスを出力します。タイマに対する3つのコンペアレジスタ設定と、ダウンカウント長設定を行うことで、それぞれ異なった波形のワンショットパルスを生成します。

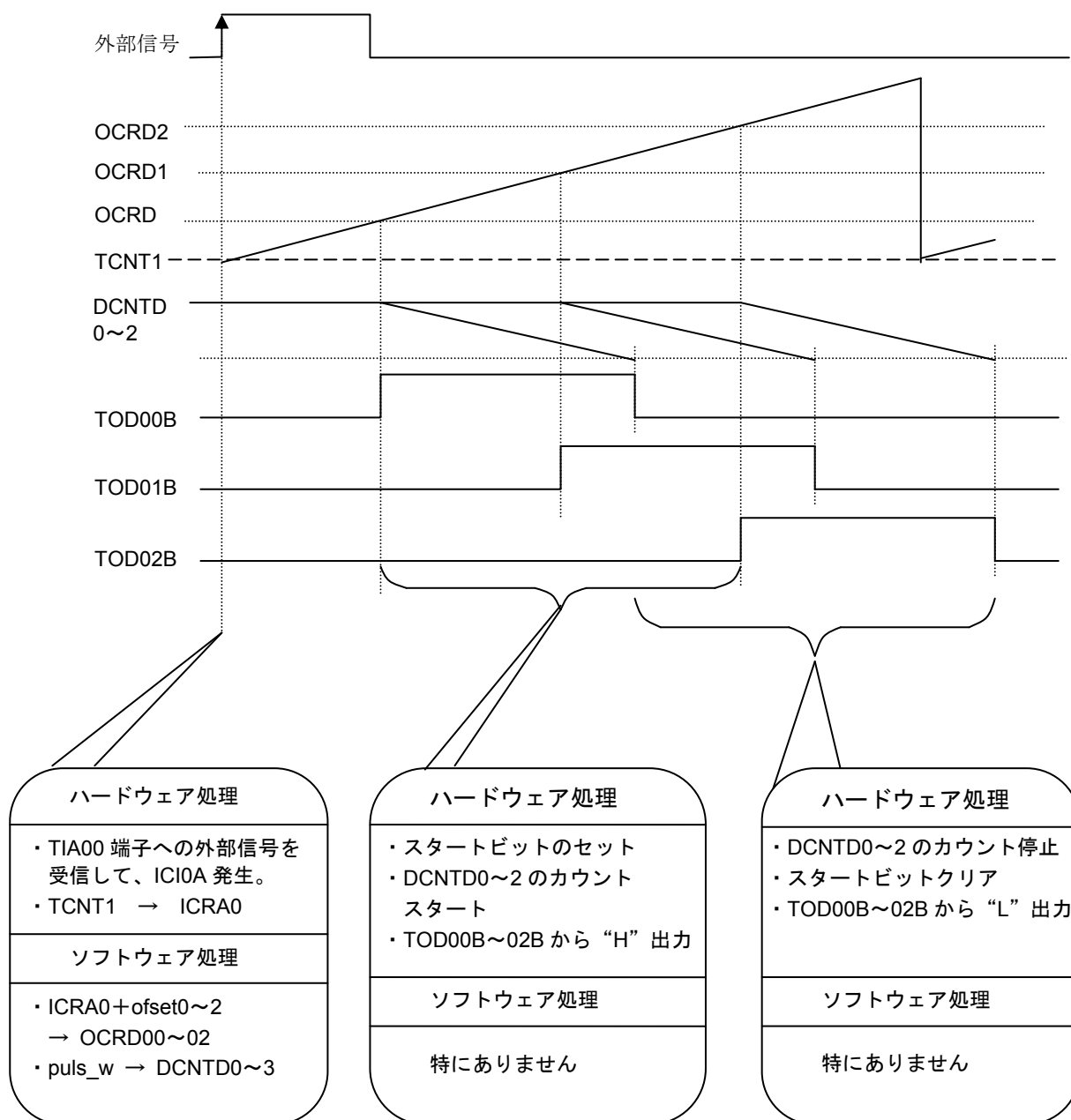


図 2.18 ワンショットパルス出力動作原理

2.10.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-Ⅲ等の初期設定を行います。
ワンショットパルス出カルーチン	ICIA00	ICFA0 によって起動し、オフセット及びパルス幅を設定し、ワンショットパルスを出力します。

● 使用定数の説明

ラベル名	機能	データ長	使用モジュール名
offset0~2	ワンショットパルスのオフセットに相当するタイマ値を設定します。オフセットは以下の式で求められます。 オフセット(ns)=タイマ値×Pφ周期(40MHz 動作時 25ns) ×10(プリスケアラの分周比)	unsigned short	メインルーチン
puls_w	ワンショットパルスのパルス幅に相当するタイマ値を設定します。パルス幅は以下の式で求められます。 パルス幅(ns)=タイマ値×Pφ周期×10	unsigned short	ワンショットパルス出力

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	PE0 端子を TIA0 に設定します。	0x0001	メインルーチン
PFCR1	PF00、01,02 を TOD 出力 00B、01B、02B に設定します。	0x0015	
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-Ⅲのタイマ A、D およびプリスケアラのカウンタ開始を設定します。	0x0013	
TIOR1A	TIA0 の立ち上がりエッジで ICRA0 ヘインプットキャプチャするように設定します。	0x0001	
TIERA	TIA0 による割り込み要求を許可します。	0x01	
ICRA0	TIA0 のインプットキャプチャで TCNTA の値を転送します。	—	
TIOR1D0	タイマ D の OCRD00~02 をコンペアマッチ要因に設定します。	0x0153F	
IPR06	TIA0 の割り込み優先レベルを 15 に設定します。	0xF000	
DCRD0	OCRD00~02 のコンペアマッチをダウンカウンタスタートトリガに設定します。	0x0222	ワンショットパルス出力
OCRD00~02	ワンショットパルスのオフセットを設定します。	—	
DCNTD00~02	ワンショットパルスのパルス幅を設定します。	—	

2.10.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : One-shot pulse output. (With offset)
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Macro definitions
48  *****/
49  #define offset0  0x2000 /* オフセット */
50  #define offset1  0x2500 /* オフセット */
51  #define offset2  0x2A00 /* オフセット */
52  #define puls_w   0x4500 /* パルス幅 */
53
54  /*****
55  Private global variables and functions
56  *****/
57
58  void main(void); /* メインルーチン */
59
60  /*****
61  * Function Name : main
62  * Description   : The main loop
63  * Arguments     : none
64  * Return Value  : none
65  *****/
66  void main(void)
67  {
68  /* PORT の設定 */
69  /* Configure PE CR1
70  b15:14 PE7MD = 0 PE7 入出力 (ポート)
71  b13:12 PE6MD = 0 PE6 入出力 (ポート)
72  b11 Reserved
73  b10 PE5MD = 0 PE5 入出力 (ポート)
74  b9 Reserved
75  b8 PE4MD = 0 PE4 入出力 (ポート)
76  b7 Reserved
77  b6 PE3MD = 0 PE3 入出力 (ポート)
78  b5:4 PE2MD = 0 PE2 入出力 (ポート)
79  b3:2 PE1MD = 0 PE1 入出力 (ポート)
80  b1 Reserved
81  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
82  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
83
84  /* Configure PF CR1
85  b15:14 PF7MD[1:0] = 00 PF7 入出力 (ポート)
86  b13:12 PF6MD[1:0] = 00 PF6 入出力 (ポート)
87  b11:10 PF5MD[1:0] = 00 PF5 入出力 (ポート)
88  b9:8 PF4MD[1:0] = 00 PF4 入出力 (ポート)
89  b7:6 PF3MD[1:0] = 00 PF3 入出力 (ポート)
90  b5:4 PF2MD[1:0] = 01 TOD02B 出力 (ATU-III)
91  b3:2 PF1MD[1:0] = 01 TOD01B 出力 (ATU-III)
92  b1:0 PF0MD[1:0] = 01 TOD00B 出力 (ATU-III) */
93  PORTF.CR1.WORD = 0x0015; /* PF0,1,2 を TOD00B, 01B, 02B 出力に設定 */
94

```

```

95  /* ATU-III Timer A の設定 */
96  /* Configure TIOR1A
97  b15:12 Reserved
98  b11:10 IOA5 = 0 TIA のインプットキャプチャ禁止
99  b9:8 IOA4 = 0 TIA のインプットキャプチャ禁止
100 b7:6 IOA3 = 0 TIA のインプットキャプチャ禁止
101 b5:4 IOA2 = 0 TIA のインプットキャプチャ禁止
102 b3:2 IOA1 = 0 TIA のインプットキャプチャ禁止
103 b1:0 IOA0 = 1 TIA の立ち上がりで ICRA にキャプチャ */
104 ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立ち上がりエッジでインプットキャプチャを行う */
105
106 /* Configure TSRA
107 b7 OVFA = 0 インプットキャプチャなし
108 b6 Reserved
109 b5 ICFA5 = 0 インプットキャプチャなし
110 b4 ICFA4 = 0 インプットキャプチャなし
111 b3 ICFA3 = 0 インプットキャプチャなし
112 b2 ICFA2 = 0 インプットキャプチャなし
113 b1 ICFA1 = 0 インプットキャプチャなし
114 b0 ICFA0 = 0 インプットキャプチャなし */
115 ATUA.TSRA.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
116
117 /* Configure TIERA
118 b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
119 b6 Reserved
120 b5 ICEA5 = 0 インプットキャプチャ割り込み A5 要求の出力を禁止
121 b4 ICEA4 = 0 インプットキャプチャ割り込み A4 要求の出力を禁止
122 b3 ICEA3 = 0 インプットキャプチャ割り込み A3 要求の出力を禁止
123 b2 ICEA2 = 0 インプットキャプチャ割り込み A2 要求の出力を禁止
124 b1 ICEA1 = 0 インプットキャプチャ割り込み A1 要求の出力を禁止
125 b0 ICEA0 = 1 インプットキャプチャ割り込み A0 要求の出力を許可 */
126 ATUA.TIERA.BYTE = 0x01; /* TIA00 のインプットキャプチャ割り込みを許可 */
127
128 /* ATU-III Timer D の設定 */
129 /* Configure TIOR1D0
130 b15:14 OSSD03[1:0] = 00 TOD03A 端子出力なし
131 b13:12 OSSD02[1:0] = 01 コンペアマッチ A 発生時、TOD02A 端子に I/O コントロールビット A で
    の設定値を出力
132 b11:10 OSSD01[1:0] = 01 コンペアマッチ A 発生時、TOD01A 端子に I/O コントロールビット A で
    の設定値を出力
133 b9:8 OSSD00[1:0] = 01 コンペアマッチ A 発生時、TOD00A 端子に I/O コントロールビット A で
    の設定値を出力
134 b7:6 IOAD03[1:0] = 00 コンペアマッチ禁止
135 b5:4 IOAD02[1:0] = 11 コンペアマッチでトグル出力
136 b3:2 IOADn1[1:0] = 11 コンペアマッチでトグル出力
137 b1:0 IOADn0[1:0] = 11 コンペアマッチでトグル出力 */
138 ATUD.SUBBLOCKB[0].TIOR1D.WORD = 0x153F; /* OCRD00, 01, 02 のコンペアマッチ A 発生で
    TOD00A, 01A, 02A はトグル出力 */
139
140 /* Configure DCRD0

```

```

141  b15 Reserved
142  b14:12 TRGSELD03[2:0] = 000 カウントスタートトリガ = カウントストップトリガ = なし
143  b11 Reserved
144  b10:8 TRGSELD02[2:0] = 010 カウントスタートトリガ = コンペアマッチ A / カウントストップ
    トリガ = なし
145  b7 Reserved
146  b6:4 TRGSELD01[2:0] = 010 カウントスタートトリガ = コンペアマッチ A / カウントストップト
    リガ = なし
147  b3 Reserved
148  b2:0 TRGSELD00[2:0] = 010 カウントスタートトリガ = コンペアマッチ A / カウントストップト
    リガ = なし */
149  ATUD.SUBBLOCKB[0].DCRD.WORD = 0x0222; /* コンペアマッチ A 発生時にダウンカウントスター
    ト */
150
151  /* Configure TSRD0
152  b15:14 Reserved
153  b13 OVFD0 = 0 TCNT2D0 でオーバフローなし
154  b12 OVFD1 = 0 TCNT1D0 でオーバフローなし
155  b11:8 UDFD03:0 = 0 アンダフローなし
156  b7:4 CMFAD03:0 = 0 コンペアマッチ A なし
157  b3:0 CMFBD03:0 = 0 コンペアマッチ B なし */
158  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンダーフローフラグをクリア
    */
159
160  /* Configure TSTRD
161  b7:4 Reserved
162  b3 STRD3 = 0 TCNT1D3、TCNT2D3、DCNTD30 のカウント動作を停止
163  b2 STRD2 = 0 TCNT1D2、TCNT2D2、DCNTD20 のカウント動作を停止
164  b1 STRD1 = 0 TCNT1D1、TCNT2D1、DCNTD10 のカウント動作を停止
165  b0 STRD0 = 1 TCNT1D0、TCNT2D0、DCNTD00 のカウント動作を許可 */
166  ATUD.TSTRD.BYTE = 0x01; /* タイマ D のサブブロック 0 のカウントをスタート */
167
168  /* Configure PSCR0
169  b15:10 Reserved
170  b9:0 PSC0[9:0] = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
171  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラ 0 の分周比を 10 に設定 */
172
173  /* Configure IPR06
174  b15:12 ATU-A(ICIA0,1) = H'F 対応する割り込みの優先順位を 15 に設定
175  b11:8 ATU-A(ICIA2,3) = H'0 対応する割り込みの優先順位を 0 に設定
176  b7:4 ATU-A(ICIA4,5) = H'0 対応する割り込みの優先順位を 0 に設定
177  b3:0 ATU-A(OVIA) = H'0 対応する割り込みの優先順位を 0 に設定 */
178  INTC.IPR06.WORD = 0xF000; /* TIA00 の割り込み(ICIA0)の優先順位を 15 に設定 */
179
180  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
181
182  /* Configure ATUENR
183  b15:10 Reserved
184  b9 TJE = 0 タイマ J のカウント動作を停止
185  b8 THE = 0 タイマ H のカウント動作を停止
186  b7 TGE = 0 タイマ G のカウント動作を停止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

187  b6 TFE = 0 タイマ F のカウント動作を停止
188  b5 TEE = 0 タイマ E のカウント動作を停止
189  b4 TDE = 0 タイマ D のカウント動作を許可
190  b3 TCE = 0 タイマ C のカウント動作を停止
191  b2 TBE = 0 タイマ B のカウント動作を停止
192  b1 TAE = 1 タイマ A のカウント動作を許可
193  b0 PSCE = 1 プリスケアラのカウント動作を許可 */
194  ATUCTRL.ATUENR.WORD = 0x0013; /* タイマ A,D およびプリスケアラのカウントをスタート */
195
196  while(1);
197  } /* End of function main() */
198
199  /*****
200  * Function Name : ICIA0
201  * Description   : インพุットキャプチャ割り込み
202  * Arguments    : none
203  * Return Value  : none
204  *****/
205  void ICIA0(void)
206  {
207  /* Configure TSRA
208  b7 OVFA = 0 インพุットキャプチャフラグをクリア
209  b6 Reserved
210  b5 ICFA5 = 0 インพุットキャプチャフラグをクリア
211  b4 ICFA4 = 0 インพุットキャプチャフラグをクリア
212  b3 ICFA3 = 0 インพุットキャプチャフラグをクリア
213  b2 ICFA2 = 0 インพุットキャプチャフラグをクリア
214  b1 ICFA1 = 0 インพุットキャプチャフラグをクリア
215  b0 ICFA0 = 0 インพุットキャプチャフラグをクリア */
216  ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
217
218  /* Configure TIERA
219  b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
220  b6 Reserved
221  b5 ICEA5 = 0 インพุットキャプチャ割り込み A5 要求の出力を禁止
222  b4 ICEA4 = 0 インพุットキャプチャ割り込み A4 要求の出力を禁止
223  b3 ICEA3 = 0 インพุットキャプチャ割り込み A3 要求の出力を禁止
224  b2 ICEA2 = 0 インพุットキャプチャ割り込み A2 要求の出力を禁止
225  b1 ICEA1 = 0 インพุットキャプチャ割り込み A1 要求の出力を禁止
226  b0 ICEA0 = 0 インพุットキャプチャ割り込み A0 要求の出力を禁止 */
227  ATUA.TIERA.BYTE = 0x00; /* TIA00 のインพุットキャプチャ割り込みを禁止 */
228
229  /* インพุットキャプチャ時のカウントに一定数を加えてオフセットの値を決定 */
230  ATUD.SUBBLOCKB[0].OCD[0] = (ATUA.ICRA0 + ofset0) << 8; /* OCD00 に TCNT1D0 に対
    応するコンペアマッチ A の条件を設定 */
231  ATUD.SUBBLOCKB[0].OCD[1] = (ATUA.ICRA0 + ofset1) << 8; /* OCD01 に TCNT1D1 に対
    応するコンペアマッチ A の条件を設定 */
232  ATUD.SUBBLOCKB[0].OCD[2] = (ATUA.ICRA0 + ofset2) << 8; /* OCD02 に TCNT1D2 に対
    応するコンペアマッチ A の条件を設定 */
233

```

```
234    /* ダウンカウンタの長さを決定し、パルスの High 幅を設定 */
235    ATUD.SUBBLOCKB[0].DCNTD[0] = puls_w << 8;
236    ATUD.SUBBLOCKB[0].DCNTD[1] = puls_w << 8;
237    ATUD.SUBBLOCKB[0].DCNTD[2] = puls_w << 8;
238    } /* End of function ICIA0() */
```

2.10.6 SH7253 グループでご使用の場合の変更点

本文 2.10 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.10.5 サンプルプログラムリストの 82 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

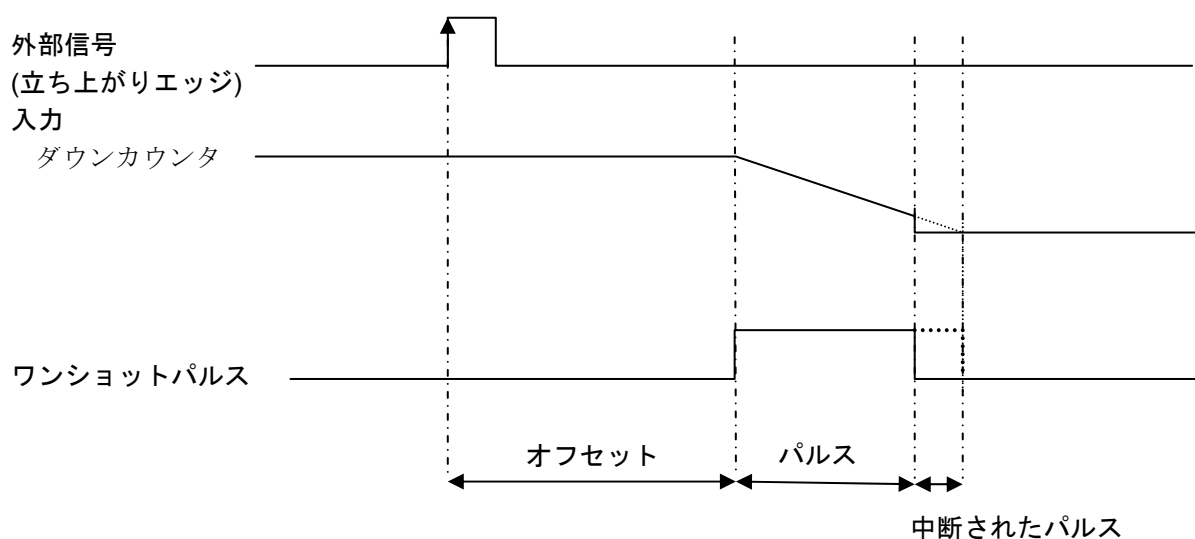
2.11 動作例 10 ワンショットパルス出力(オフセット、ターミート付)[タイマ D]

2.11.1 概要

- 1) 図 2.19に示すように外部信号の立ち上がりに同期してワンショットパルスを出力します。オフセット、パルス幅は内部クロックカウント値を設定します。
- 2) DCRD への設定によりパルス出力を強制的に終了しパルス幅を制御します。
- 3) 本タスク例では、外部信号の立ち上がりからのオフセット及びパルス幅は以下に示す範囲で可変可能です。

$$4\mu\text{s} < \text{オフセット} < P\phi \text{ 周期} \times 65536$$

$$\text{*注: } 200\text{ns} \leq \text{パルス幅} < 13\text{ms}$$



*注：オフセットは外部信号の周期より小さく、
オフセット設定ルーチンの実行時間(約 $4\mu\text{s}$)より大きいこと。

図 2.19 ワンショットパルス出力

2.11.2 使用機能説明

表 2.10に使用端子及び関連レジスタの機能割り付けを示します。

表 2.10 ATU-III機能割り付け

使用端子		機能
端子	TIA00	開始用の外部信号を入力します。
	TOD00~02B	ワンショットパルスを出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ A レジスタ	TIOR1A	キャプチャする外部入力のエッジの向きを設定する。
	TSRA	外部入力のインプットキャプチャが発生した場合、フラグがセットされます。
	TIERA	TIA00 のインプットキャプチャ割り込みを許可します。
タイマ D レジスタ	TIOR1D	OCRD に対応するコンペアマッチの許可、および発生時の動作の設定を行います。
	TIOR2D	GRD の機能の設定を行います。
	DCRD	ダウンカウントの開始、停止条件を設定します。
	TSRD	コンペアマッチ及び、オーバー、アンダー両フローが発生した場合、フラグがセットされます。
	TSTRD	タイマ D のサブブロックのカウント動作を設定します。
INTC レジスタ	IPR06	TIA00 のコンペアマッチ割り込みの優先度を設定します。

2.11.3 動作説明

図 2.20に動作原理を示します。図に示すように SH72546R のハードウェア及びソフトウェアの処理によりワンショットパルスを出力します。ダウンカウンタの長さを調整することで、ワンショットパルス出力の強制終了(L レベル出力)までの期間を設定します。

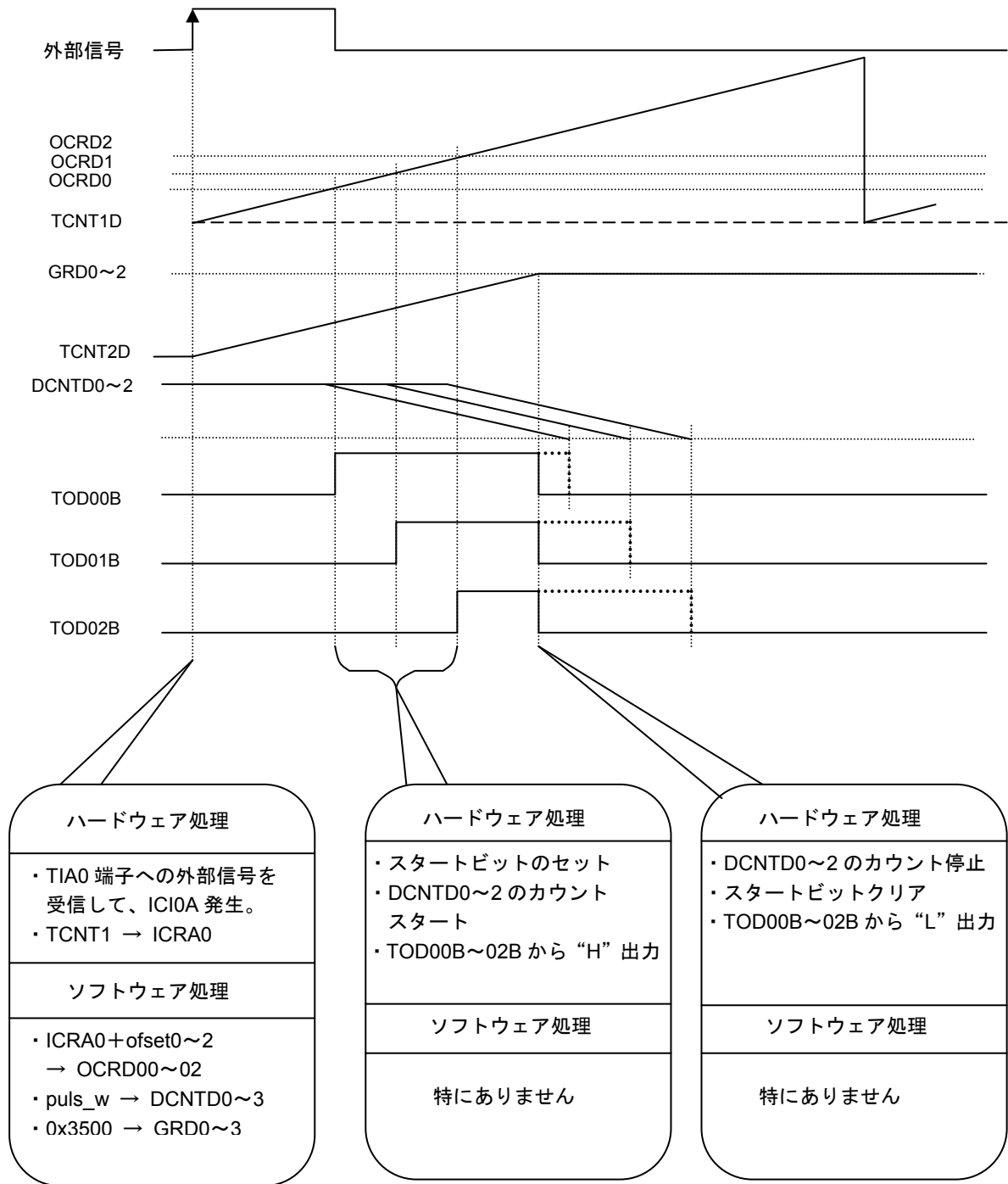


図 2.20 ワンショットパルス出力動作原理

2.11.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-III等の初期設定を行います。
ワンショットパルス出力	ICIA00	ICFA0 によって起動し、オフセット期間、ターミネート期間、およびパルス幅を設定し、ワンショットパルスを出力します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール名
Offset0~2	ワンショットパルスのオフセットに相当するタイマ値を設定します。オフセットは以下の式で求められます。 オフセット(ns)=タイマ値×Pφ周期(40MHz 動作時 25ns) ×10(プリスケアラの分周比)	unsigned short	全モジュールで使用
Puls_w	ワンショットパルスのパルス幅に相当するタイマ値を設定します。パルス幅は以下の式で求められます。 パルス幅(ns)=タイマ値×Pφ周期×10	unsigned short	

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	PE0 端子を TIA0 に設定します。	0x0001	メインルーチン
PFCR1	PF00~02 端子を TOD 出力 00B、01B、02B に設定します。	0x0015	
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIのタイマ A、D およびプリスケアラのカウント開始を設定します。	0x0013	
TIOR1A	TIA0 の立ち上がりエッジで ICRA0 へインプットキャプチャするように設定します。	0x0001	
TIERA	TIA0 による割り込み要求を許可します。	0x01	
ICRA0	TIA0 のインプットキャプチャで TCNTA の値を転送します。	—	
TIOR1D0	TCNT1D と OCRD00~02 のコンペアマッチを許可します。	0x153F	
TIOR2D0	TCNT2D と GRD00~02 のコンペアマッチを許可します。	0x0333	
DCRD0	OCRD00~02 のコンペアマッチをダウンカウンタスタートトリガに、GED00~02 のコンペアマッチをダウンカウンタストップトリガに、それぞれ設定します。	0x0333	
OCRD00~02	ワンショットパルスのオフセットを設定します。	—	ワンショットパルス出力
GRD00~02	ワンショットパルスのターミネートまでの期間を設定します。	—	
DCNTD00~02	ワンショットパルスのパルス幅を設定します。	—	
IPR06	TIA0 の割り込み優先レベルを 15 に設定します。	0xF000	メインルーチン

2.11.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : One-shot pulse output. (Offset, with termination)
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Macro definitions
48  *****/
49  #define Offset0  0x2000 /* オフセット */
50  #define Offset1  0x2500 /* オフセット */
51  #define Offset2  0x2A00 /* オフセット */
52  #define Puls_w   0x2500 /* パルス幅 */
53
54  /*****
55  Private global variables and functions
56  *****/
57
58  void main(void); /* メインルーチン */
59
60  /*****
61  * Function Name : main
62  * Description   : The main loop
63  * Arguments     : none
64  * Return Value  : none
65  *****/
66  void main(void)
67  {
68  /* PORT の設定 */
69  /* Configure PE CR1
70  b15:14 PE7MD = 0 PE7 入出力 (ポート)
71  b13:12 PE6MD = 0 PE6 入出力 (ポート)
72  b11 Reserved
73  b10 PE5MD = 0 PE5 入出力 (ポート)
74  b9 Reserved
75  b8 PE4MD = 0 PE4 入出力 (ポート)
76  b7 Reserved
77  b6 PE3MD = 0 PE3 入出力 (ポート)
78  b5:4 PE2MD = 0 PE2 入出力 (ポート)
79  b3:2 PE1MD = 0 PE1 入出力 (ポート)
80  b1 Reserved
81  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
82  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA0 入力に設定 */
83
84  /* Configure PF CR1
85  b15:14 PF7MD[1:0] = 00 PF7 入出力 (ポート)
86  b13:12 PF6MD[1:0] = 00 PF6 入出力 (ポート)
87  b11:10 PF5MD[1:0] = 00 PF5 入出力 (ポート)
88  b9:8 PF4MD[1:0] = 00 PF4 入出力 (ポート)
89  b7:6 PF3MD[1:0] = 00 PF3 入出力 (ポート)
90  b5:4 PF2MD[1:0] = 01 TOD02B 出力 (ATU-III)
91  b3:2 PF1MD[1:0] = 01 TOD01B 出力 (ATU-III)
92  b1:0 PF0MD[1:0] = 01 TOD00B 出力 (ATU-III) */
93  PORTF.CR1.WORD = 0x0015; /* PF0,1,2 を TOD00B, 01B, 02B 出力に設定 */
94

```

```

95  /* ATU-III Timer A の設定 */
96  /* Configure TIOR1A
97  b15:12 Reserved
98  b11:10 IOA5 = 0 TIA のインプットキャプチャ禁止
99  b9:8 IOA4 = 0 TIA のインプットキャプチャ禁止
100 b7:6 IOA3 = 0 TIA のインプットキャプチャ禁止
101 b5:4 IOA2 = 0 TIA のインプットキャプチャ禁止
102 b3:2 IOA1 = 0 TIA のインプットキャプチャ禁止
103 b1:0 IOA0 = 1 TIA の立ち上がりで ICRA にキャプチャ */
104 ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立ち上がりエッジで ICRA にインプットキャプチャを行う
   */
105
106 /* Configure TSRA
107 b7 OVFA = 0 インプットキャプチャフラグをクリア
108 b6 Reserved
109 b5 ICFA5 = 0 インプットキャプチャフラグをクリア
110 b4 ICFA4 = 0 インプットキャプチャフラグをクリア
111 b3 ICFA3 = 0 インプットキャプチャフラグをクリア
112 b2 ICFA2 = 0 インプットキャプチャフラグをクリア
113 b1 ICFA1 = 0 インプットキャプチャフラグをクリア
114 b0 ICFA0 = 0 インプットキャプチャフラグをクリア */
115 ATUA.TSRA.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
116
117 /* Configure TIERA
118 b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
119 b6 Reserved
120 b5 ICEA5 = 0 インプットキャプチャ割り込み A5 要求の出力を禁止
121 b4 ICEA4 = 0 インプットキャプチャ割り込み A4 要求の出力を禁止
122 b3 ICEA3 = 0 インプットキャプチャ割り込み A3 要求の出力を禁止
123 b2 ICEA2 = 0 インプットキャプチャ割り込み A2 要求の出力を禁止
124 b1 ICEA1 = 0 インプットキャプチャ割り込み A1 要求の出力を禁止
125 b0 ICEA0 = 1 インプットキャプチャ割り込み A0 要求の出力を許可 */
126 ATUA.TIERA.BYTE = 0x01; /* TIA00 のインプットキャプチャ割り込みを許可 */
127
128 /* ATU-III Timer D の設定 */
129 /* Configure TIOR1D0
130 b15:14 OSSD03[1:0] = 00 TOD03A 端子出力なし
131 b13:12 OSSD02[1:0] = 01 コンペアマッチ A 発生時、TOD02A 端子に I/O コントロールビット A で
   の設定値を出力
132 b11:10 OSSD01[1:0] = 01 コンペアマッチ A 発生時、TOD01A 端子に I/O コントロールビット A で
   の設定値を出力
133 b9:8 OSSD00[1:0] = 01 コンペアマッチ A 発生時、TOD00A 端子に I/O コントロールビット A で
   の設定値を出力
134 b7:6 IOAD03[1:0] = 00 コンペアマッチ禁止
135 b5:4 IOAD02[1:0] = 11 コンペアマッチでトグル出力
136 b3:2 IOADn1[1:0] = 11 コンペアマッチでトグル出力
137 b1:0 IOADn0[1:0] = 11 コンペアマッチでトグル出力 */
138 ATUD.SUBBLOCKB[0].TIOR1D.WORD = 0x153F; /* OCRD00, 01, 02 のコンペアマッチ A 発生で
   TOD00A, 01A, 02A はトグル出力 */
139

```

```

140  /* Configure TIOR2D0
141  b15 Reserved
142  b14:12 IOBDn3[2:0] = 000 コンペアマッチ禁止 (GRDn3 をアウトプットコンペアレジスタとして
    利用するとき)
143  b11 Reserved
144  b10:8 IOBDn2[2:0] = 011 コンペアマッチでトグル出力 (GRDn2 をアウトプットコンペアレジスタ
    として利用するとき)
145  b7 Reserved
146  b6:4 IOBDn1[2:0] = 011 コンペアマッチでトグル出力 (GRDn1 をアウトプットコンペアレジスタと
    して利用するとき)
147  b3 Reserved
148  b2:0 IOBDn0[2:0] = 011 コンペアマッチでトグル出力 (GRDn0 をアウトプットコンペアレジスタと
    して利用するとき) */
149  ATUD.SUBBLOCKB[0].TIOR2D.WORD=0x0333; /* GRD00, 01, 02 のコンペアマッチ B 発生で TOD00A
    はトグル出力 */
150
151  /* Configure DCRD0
152  b15 Reserved
153  b14:12 TRGSELD03[2:0] = 000 カウントスタートトリガ = カウントストップトリガ = なし
154  b11 Reserved
155  b10:8 TRGSELD02[2:0] = 011 カウントスタートトリガ = コンペアマッチ A / カウントストップト
    リガ = コンペアマッチ B
156  b7 Reserved
157  b6:4 TRGSELD01[2:0] = 011 カウントスタートトリガ = コンペアマッチ A / カウントストップト
    リガ = コンペアマッチ B
158  b3 Reserved
159  b2:0 TRGSELD00[2:0] = 011 カウントスタートトリガ = コンペアマッチ A / カウントストップト
    リガ = コンペアマッチ B */
160  ATUD.SUBBLOCKB[0].DCRD.WORD = 0x0333; /* コンペアマッチ A 発生時にダウンカウントスター
    ト、コンペアマッチ B 発生時にダウンカウントストップ */
161
162  /* Configure TSRD0
163  b15:14 Reserved
164  b13 OVF2D0 = 0 オーバフローフラグをクリア
165  b12 OVF1D0 = 0 オーバフローフラグをクリア
166  b11:8 UDFD03:0 = 0 アンダフローフラグをクリア
167  b7:4 CMFAD03:0 = 0 コンペアマッチ A フラグをクリア
168  b3:0 CMFBD03:0 = 0 コンペアマッチ B フラグをクリア */
169  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンダーフローフラグをクリア
    */
170
171  /* Configure TSTRD
172  b7:4 Reserved
173  b3 STRD3 = 0 TCNT1D3、TCNT2D3、DCNTD30 のカウント動作を停止
174  b2 STRD2 = 0 TCNT1D2、TCNT2D2、DCNTD20 のカウント動作を停止
175  b1 STRD1 = 0 TCNT1D1、TCNT2D1、DCNTD10 のカウント動作を停止
176  b0 STRD0 = 1 TCNT1D0、TCNT2D0、DCNTD00 のカウント動作を許可 */
177  ATUD.TSTRD.BYTE = 0x01; /* タイマ D のサブブロック 0 のカウントをスタート */
178
179  /* Configure PSCR0

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

180  b15:10 Reserved
181  b9:0 PSC0[9:0] = H'9 プリスケアラの分周比を 10(設定値は-1)に設定 */
182  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケアラ 0 の分周比を 10 に設定 */
183
184  /* Configure IPR06
185  b15:12 ATU-A(ICIA0,1) = H'F 対応する割り込みの優先順位を設定
186  b11:8 ATU-A(ICIA2,3) = H'0 対応する割り込みの優先順位を設定
187  b7:4 ATU-A(ICIA4,5) = H'0 対応する割り込みの優先順位を設定
188  b3:0 ATU-A(OVIA) = H'0 対応する割り込みの優先順位を設定 */
189  INTC.IPR06.WORD = 0xF000; /* TIA00 の割り込み(ICIA0)の優先順位を 15 に設定 */
190
191  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
192
193  /* Configure ATUENR
194  b15:10 Reserved
195  b9 TJE = 0 タイマ J のカウント動作を停止
196  b8 THE = 0 タイマ H のカウント動作を停止
197  b7 TGE = 0 タイマ G のカウント動作を停止
198  b6 TFE = 0 タイマ F のカウント動作を停止
199  b5 TEE = 0 タイマ E のカウント動作を停止
200  b4 TDE = 0 タイマ D のカウント動作を許可
201  b3 TCE = 0 タイマ C のカウント動作を停止
202  b2 TBE = 0 タイマ B のカウント動作を停止
203  b1 TAE = 1 タイマ A のカウント動作を許可
204  b0 PSCE = 1 プリスケアラのカウント動作を許可 */
205  ATUCTRL.ATUENR.WORD = 0x0013; /* タイマ A, D およびプリスケアラのカウントをスタート */
206
207  while(1);
208  } /* End of function main() */
209
210  /*****
211  * Function Name : ICIA0
212  * Description    : インพุットキャプチャ割り込み
213  * Arguments      : none
214  * Return Value   : none
215  *****/
216  void ICIA0(void)
217  {
218  /* Configure TSRA
219  b7 OVFA = 0 インพุットキャプチャなし
220  b6 Reserved
221  b5 ICFA5 = 0 インพุットキャプチャなし
222  b4 ICFA4 = 0 インพุットキャプチャなし
223  b3 ICFA3 = 0 インพุットキャプチャなし
224  b2 ICFA2 = 0 インพุットキャプチャなし
225  b1 ICFA1 = 0 インพุットキャプチャなし
226  b0 ICFA0 = 0 インพุットキャプチャなし */
227  ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャをクリア */
228
229  /* Configure TIERA

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

230  b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
231  b6 Reserved
232  b5 ICEA5 = 0 インพุットキャプチャ割り込み A5 要求の出力を禁止
233  b4 ICEA4 = 0 インพุットキャプチャ割り込み A4 要求の出力を禁止
234  b3 ICEA3 = 0 インพุットキャプチャ割り込み A3 要求の出力を禁止
235  b2 ICEA2 = 0 インพุットキャプチャ割り込み A2 要求の出力を禁止
236  b1 ICEA1 = 0 インพุットキャプチャ割り込み A1 要求の出力を禁止
237  b0 ICEA0 = 0 インพุットキャプチャ割り込み A0 要求の出力を禁止 */
238  ATUA.TIERA.BYTE = 0x00; /* TIA00 のインพุットキャプチャ割り込みを禁止 */
239
240  /* インพุットキャプチャ時のカウントに一定数を加えてオフセットの値を決定 */
241  ATUD.SUBBLOCKB[0].OCR0D[0] = (ATUA.ICRA0 + Offset0) << 8; /* OCR0D00、TCNT1D0 に対
    応するコンペアマッチ A の条件を設定 */
242  ATUD.SUBBLOCKB[0].OCR0D[1] = (ATUA.ICRA0 + Offset1) << 8; /* OCR0D01、TCNT1D1 に対
    応するコンペアマッチ A の条件を設定 */
243  ATUD.SUBBLOCKB[0].OCR0D[2] = (ATUA.ICRA0 + Offset2) << 8; /* OCR0D02、TCNT1D2 に対
    応するコンペアマッチ A の条件を設定 */
244
245  /* ダウンカウントの長さを決定し、ターミネートの実行時間を設定 */
246  ATUD.SUBBLOCKB[0].GRD[0] = (ATUA.ICRA0 + 0x3500) << 8; /* GRD000、TCNT2D0 に対応す
    るコンペアマッチ B の条件を設定 */
247  ATUD.SUBBLOCKB[0].GRD[1] = (ATUA.ICRA0 + 0x3500) << 8; /* GRD001、TCNT2D1 に対応す
    るコンペアマッチ B の条件を設定 */
248  ATUD.SUBBLOCKB[0].GRD[2] = (ATUA.ICRA0 + 0x3500) << 8; /* GRD002、TCNT2D2 に対応す
    るコンペアマッチ B の条件を設定 */
249
250  /* ダウンカウントの長さを決定し、パルスの High 幅を設定 */
251  ATUD.SUBBLOCKB[0].DCNTD[0] = Puls_w << 8;
252  ATUD.SUBBLOCKB[0].DCNTD[1] = Puls_w << 8;
253  ATUD.SUBBLOCKB[0].DCNTD[2] = Puls_w << 8;
254  } /* End of function ICIA0() */

```


2.11.6 SH7253 グループでご使用の場合の変更点

本文 2.11 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.11.5 サンプルプログラムリストの 82 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.12 動作例 11 PWM 波形出力[タイマ E]

2.12.1 概要

図 2.21に示すように、デューティ及び周期を変化できるパルスを出力します。タイマ E のサイクルレジスタとデューティレジスタの設定を行うことで、パルス幅を設定します。

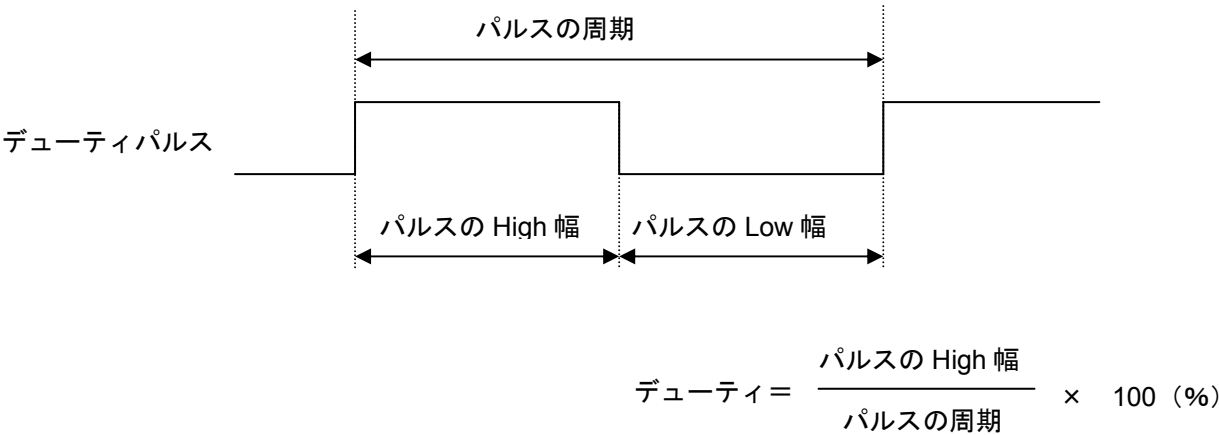


図 2.21 PWM 出力の概要図

2.12.2 使用機能説明

表 2.11に使用端子及び関連レジスタの機能割り付けを示します。

表 2.11 ATU-III機能割り付け

使用端子		機能
端子	TOE00	PWM 波形を出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ E レジスタ	DTRE00	デューティレジスタの値を設定します。
	DRLDE00	デューティリロードレジスタの値を設定します。
	CYLRE00	サイクルレジスタの値を設定します。
	CRLDE00	サイクルリロードレジスタの値を設定します。
	RLDCRE0	DTRE のリロード機能を有効に設定します。
	TIERE0	サイクルマッチ割り込みを設定します。
	TSTRE	タイマ E 各サブブロックのカウンタの動作を設定します。
	SSTRE0	タイマ E サブブロック 0 のカウンタ動作を許可します。
PFC レジスタ	PECR1	端子機能を設定します。
INTC レジスタ	IPR14	サイクルマッチ割り込みの優先度を設定します。

2.12.3 動作説明

図 2.22に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって PWM 波形を出力します。コンペアマッチの度にデューティリロードレジスタの値を更新することで、PWM 波形の Duty 比を変化させています。

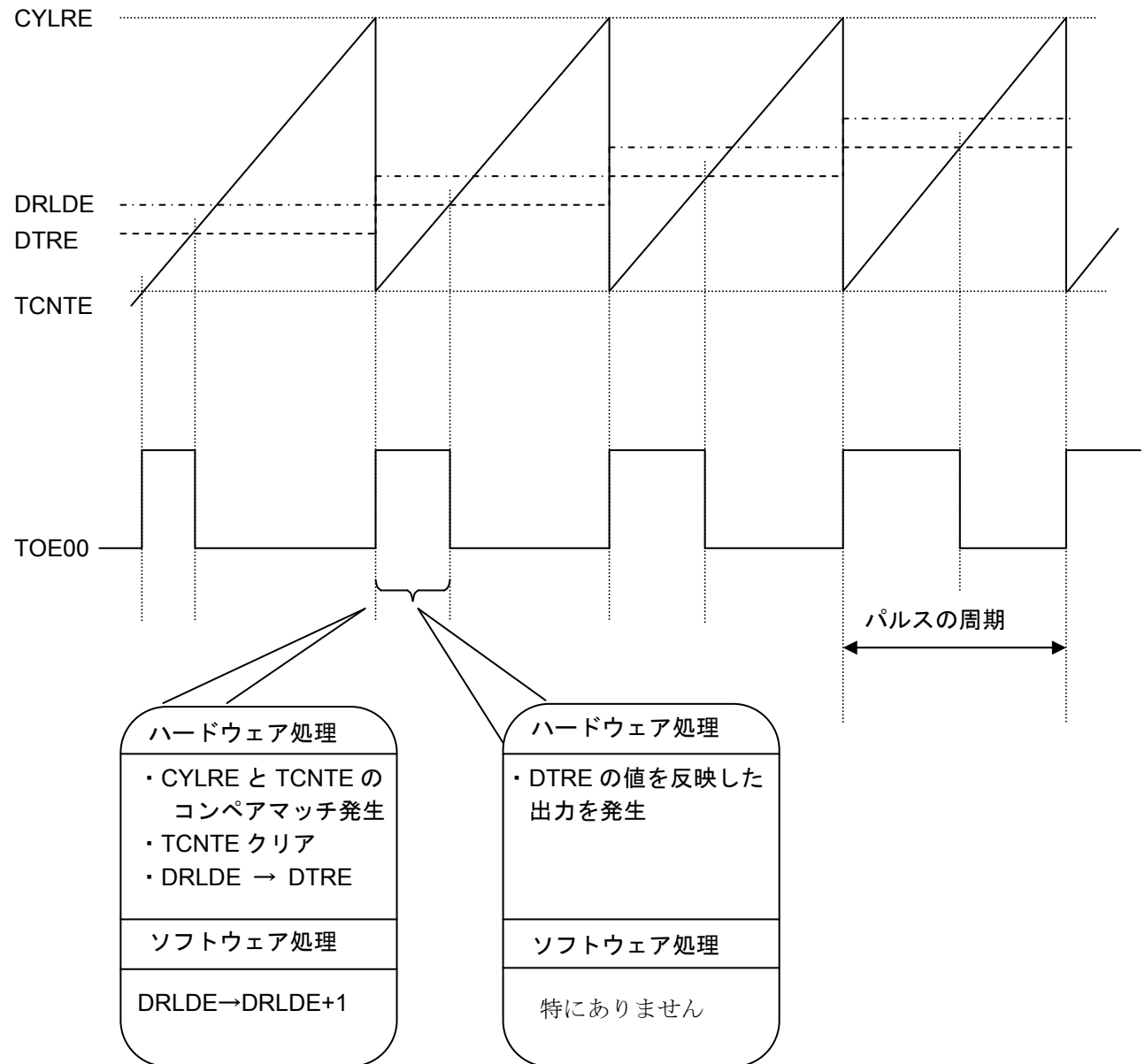


図 2.22 PWM 出力動作原理

2.12.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
サイクルマッチ 割り込み	CMIE00	CMFE00 によって起動し、DRLDE の値を増減します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール名
up	サイクルマッチ割り込み発生時、デューティリロードレジスタの値が一定 以下/以上でセット/クリアされ、デューティリロードレジスタ値の増減方向を設定するフラグになります。	unsigned char	サイクルマッチ 割り込み

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	PE6 端子を TOE00 に設定します。	0x1000	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 2 に設定します。	0x0001	
ATUENR	ATU-IIIのタイマ E およびプリスケアラのカウント開始を許可します。	0x0021	
TSTRE	タイマ E サブブロック 0 の動作を許可します。	0x01	
SSTRE0	タイマ E サブブロック 0 タイマカウンタ 00 の動作を許可します。	0x01	
TIERE0	サイクルレジスタ 00 によるサイクルマッチ割り込みを許可する。	0x01	
RLDCRE0	DTRE00 のリロード機能を有効に設定する。	0x01	
DTRE00	PWM 波形のデューティを設定します。	0x0001	
CYLRE00	PWM 波形の周期を設定します。	0x0100	
DRLDE00	サイクルマッチ割り込み発生時にこのレジスタの値を DTRE にセットします。	0x0001	サイクルマッチ 割り込み
CRLDE00	サイクルマッチ割り込み発生時にこのレジスタの値を CYLRE にセットします。	0x0100	
IPR14	CMIE0 の割り込み優先レベルを 15 に設定します。	0x00F0	メインルーチン

2.12.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : PWM wave output.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned char up;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58  void main(void)
59  {
60  up = 0;
61
62  /* ポートの設定 */
63  /* Configure PECR1
64  b15:14 PE7MD = 00 PE7 入出力 (ポート)
65  b13:12 PE6MD = 01 TOE00 出力 (ATU-III)
66  b11 Reserved
67  b10 PE5MD = 0 PE5 入出力 (ポート)
68  b9 Reserved
69  b8 PE4MD = 0 PE4 入出力 (ポート)
70  b7 Reserved
71  b6 PE3MD = 0 PE3 入出力 (ポート)
72  b5:4 PE2MD = 00 PE2 入出力 (ポート)
73  b3:2 PE1MD = 00 PE1 入出力 (ポート)
74  b1 Reserved
75  b0 PE0MD = 0 PE0 入出力 (ポート) */
76  PORTE.CR1.WORD = 0x1000; /* PE6 を TOE00 出力に設定 */
77
78  /* ATU-IIIの設定 */
79  /* Configure DTRE00
80  b15-0 DTRENm[15:0] PWM のデューティ */
81  ATUE.SUBBLOCK[0].DTRE[0] = 0x0001; /* デューティレジスタの値を 0x0001 に設定 */
82
83  /* Configure DRLDE00
84  b15-0 DRLDE [15:0] PWM のデューティ(リロード用) */
85  ATUE.SUBBLOCK[0].DRLDE[0] = 0x0001; /* デューティリロードレジスタ値を 0x01 に設定 */
86
87  /* Configure CYLRE00
88  b15-0 CRLDE00[15:0] PWM の周期 */
89  ATUE.SUBBLOCK[0].CYLRE[0] = 0x0100; /* サイクルレジスタの値を 0x00FF に設定 */
90
91  /* Configure DRLDE00
92  b15-0 CYLRE00[15:0] PWM の周期(リロード用) */
93  ATUE.SUBBLOCK[0].CRLDE[0] = 0x0100; /* サイクルリロードレジスタの値を 0x00FF に設定 */
94

```

```

95  /* Configure RLDCRE0
96  b7:4 Reserved
97  b3 RLDENen3 = 0 サイクルマッチ時のデューティレジスタ、サイクルレジスタのリロード機能を無効
    に設定
98  b2 RLDENen2 = 0 サイクルマッチ時のデューティレジスタ、サイクルレジスタのリロード機能を無効
    に設定
99  b1 RLDENen1 = 0 サイクルマッチ時のデューティレジスタ、サイクルレジスタのリロード機能を無効
    に設定
100 b0 RLDENen0 = 1 サイクルマッチ時のデューティレジスタ、サイクルレジスタのリロード機能を有効
    に設定 */
101 ATUE.SUBBLOCK[0].RLDCRE.BYTE = 0x01; /* DTRE00 のリロード機能を有効に設定 */
102
103 /* Configure TIERE0
104 b7:4 Reserved
105 b3 CMEE03 = 0 CMFE03 による割り込み要求を禁止
106 b2 CMEE02 = 0 CMFE02 による割り込み要求を禁止
107 b1 CMEE01 = 0 CMFE01 による割り込み要求を禁止
108 b0 CMEE00 = 1 CMFE00 による割り込み要求を許可 */
109 ATUE.SUBBLOCK[0].TIERE.BYTE = 0x01; /* サイクルレジスタ 00 のサイクルマッチ発生による割
    り込みを許可 */
110
111 /* Configure TSTRE
112 b7:6 Reserved
113 b5 STRE5 = 0 サブブロック E5 の動作を禁止
114 b4 STRE4 = 0 サブブロック E4 の動作を禁止
115 b3 STRE3 = 0 サブブロック E3 の動作を禁止
116 b2 STRE2 = 0 サブブロック E2 の動作を禁止
117 b1 STRE1 = 0 サブブロック E1 の動作を禁止
118 b0 STRE0 = 1 サブブロック E0 の動作を許可 */
119 ATUE.TSTRE.BYTE = 0x01; /* タイマ E サブブロック 0 の動作を許可 */
120
121 /* Configure SSTRE0
122 b7:4 Reserved
123 b3 SSTRE03 = 0 サブブロック E0 のチャンネル 3 のカウント動作を停止
124 b2 SSTRE02 = 0 サブブロック E0 のチャンネル 2 のカウント動作を停止
125 b1 SSTRE01 = 0 サブブロック E0 のチャンネル 1 のカウント動作を停止
126 b0 SSTRE00 = 1 サブブロック E0 のチャンネル 0 のカウント動作を許可 */
127 ATUE.SUBBLOCK[0].SSTRE.BYTE = 0x01; /* タイマ E サブブロック 0 カウンタ E0 の動作を許可 */
128
129 /* Configure PSCR0
130 b15:10 Reserved
131 b9:0 PSC0[9:0] = H'1 プリスケーラの分周比に 2(設定値は-1)を設定 */
132 ATUCTRL.PSCR0.WORD = 0x0001; /* プリスケーラの分周比を 2 に設定 */
133
134 /* Configure IPR14
135 b15:12 Reserved
136 b11:8 Reserved
137 b7:4 ATU-E0(CMIE00:03) = H'F 対応する割り込みの優先順位を 15 に設定
138 b3:0 ATU-E1(CMIE10-13) = H'0 対応する割り込みの優先順位を 0 に設定 */
139 INTC.IPR14.WORD = 0x00F0; /* CMIE00 の割り込み優先順位を 15 に設定 */
140

```

```

141  set_imask(0); /* 割り込みマスキレベルを 0 に設定 */
142
143  /* Configure ATUENR
144  b15:10 Reserved
145  b9  TJE = 0 タイマ J のカウント動作を停止
146  b8  THE = 0 タイマ H のカウント動作を停止
147  b7  TGE = 0 タイマ G のカウント動作を停止
148  b6  TFE = 0 タイマ F のカウント動作を停止
149  b5  TEE = 1 タイマ E のカウント動作を許可
150  b4  TDE = 0 タイマ D のカウント動作を停止
151  b3  TCE = 0 タイマ C のカウント動作を停止
152  b2  TBE = 0 タイマ B のカウント動作を停止
153  b1  TAE = 0 タイマ A のカウント動作を停止
154  b0  PSCE = 1 プリスケアラのカウント動作を許可 */
155  ATUCTRL.ATUENR.WORD = 0x0021; /* タイマ E およびプリスケアラのカウントをスタート */
156
157  while(1);
158  } /* End of function main() */
159
160  /*****
161  * Function Name : CMIE00
162  * Description   : サイクルマッチ割り込み
163  * Arguments    : none
164  * Return Value  : none
165  *****/
166  void CMIE00(void)
167  {
168  /* Configure TSRE0
169  b7  OVFE03 = 0 オーバフローフラグをクリア
170  b6  OVFE02 = 0 オーバフローフラグをクリア
171  b5  OVFE01 = 0 オーバフローフラグをクリア
172  b4  OVFE00 = 0 オーバフローフラグをクリア
173  b3  CMFE03 = 0 サイクルマッチフラグをクリア
174  b2  CMFE02 = 0 サイクルマッチフラグをクリア
175  b1  CMFE01 = 0 サイクルマッチフラグをクリア
176  b0  CMFE00 = 0 サイクルマッチフラグをクリア */
177  ATUE.SUBBLOCK[0].TSRE.BYTE &= 0x00; /* サイクルマッチフラグをクリア */
178
179  if(up == 0) /* up フラグがセットされていないならば */
180  {
181  ATUE.SUBBLOCK[0].DRLDE[0]--; /* デューティサイクルレジスタの値を減少 */
182  }
183  else /* up フラグがセットされているならば */
184  {
185  ATUE.SUBBLOCK[0].DRLDE[0]++; /* デューティサイクルレジスタの値を増加 */
186  }
187
188  if(ATUE.SUBBLOCK[0].DRLDE[0] >= 0x00FF) /* デューティサイクルレジスタの値が 0x00FC 以上ならば */
189  {

```



```
190  up = 0; /* up フラグをクリア */
191  }
192  if(ATUE.SUBBLOCK[0].DRLDE[0] <= 0x0001) /* デューティサイクルレジスタの値が 0x0005 以
    下ならば */
193  {
194  up = 1; /* up フラグをセット */
195  }
196  } /* End of function CMIE00() */
```

2.12.6 SH7253 グループでご使用の場合の変更点

本文 2.12 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TOE00 端子が、SH7253 グループでは、PB4 端子（SH7254R グループでは、PE6 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.12.5 サンプルプログラムリストの 76 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PEC1 を 0x1000 に設定（変更前）

SH7253 グループ : PBCR2A を 0x0001 に設定（変更後）

2.13 動作例 12 有効エッジ入力間隔[タイマ F]

2.13.1 概要

図 2.23に示すように、設定回数の有効なエッジが入力されるまでの時間を計測します。タイマの動作モードを有効エッジ入力間隔計測に設定することで、有効エッジ数を設定します。

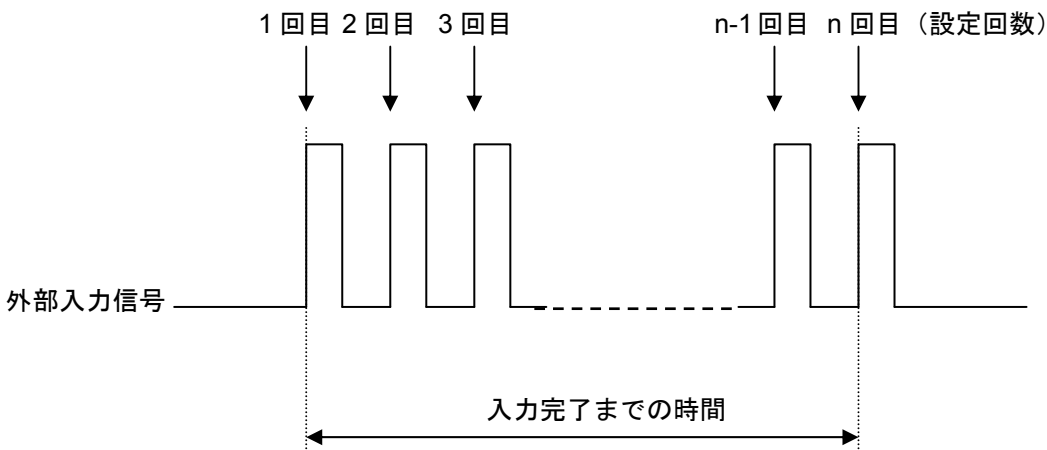


図 2.23 有効エッジ入力間隔のタイミング（立ち上がりエッジのカウント時）

2.13.2 使用機能説明

表 2.12に使用端子及び関連レジスタの機能割り付けを示します。

表 2.12 ATU-III機能割り付け

使用端子		機能
端子	TIF0A	外部から信号を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TIERF0	ICIF0 割り込みを許可します。
	TCRF0	タイマ F 各サブブロックの動作を設定します。
	GRBF0	検出するエッジの回数を設定します。
	ECNTBF0	有効な外部信号の回数を記録します。
	TSRF0	タイマ F のフラグ状態が格納されます。
	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
INTC レジスタ	IPR16	ICIF0 の割り込みの優先度を設定します。
PFC レジスタ	PHCR	端子機能を設定します。

2.13.3 動作説明

図 2.24に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって有効エッジ入力の時間を計測します。本例では検出するエッジの回数を 5 回に設定し、5 回検出後にカウンタの値を記録しています。

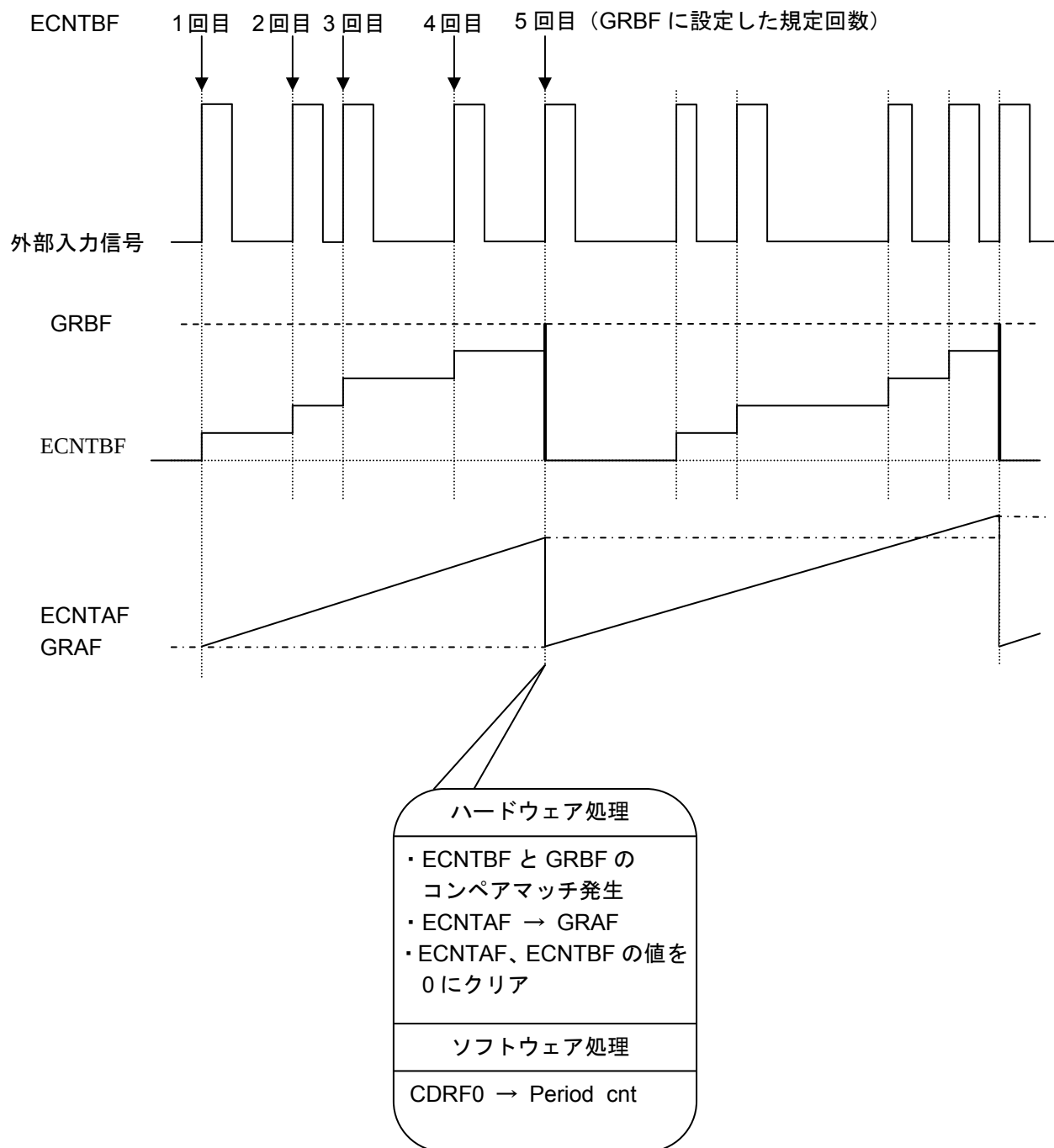


図 2.24 有効エッジ入力間隔計測動作原理

2.13.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
計測値の保存ルーチン	ICIF0	計測した値を変数に保存します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
Period_cnt	計測した規定数のエッジ入力に要した期間を保存します。	unsigned long	計測値の保存ルーチン

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PHCR	PH0 端子を TIF0A に設定します。	0x0003	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIのタイマ F およびプリスケアラのカウント開始を許可します。	0x0041	
TIERF0	割り込みを許可します。	0x01	
IPR16	割り込みの優先順位を設定します。	0xF000	
TSTRF	タイマ F サブブロック 0 のカウントをスタートさせる。	0x00000001	
TCRF0	サブブロック 0 を「クロックバス 0 でカウント」、「有効エッジ入力間隔計測」「計測エッジは立ち上がり」に設定します。	0x05	
GRBF0	検出するエッジの回数を 5 回に設定します。	0x0005	
ECNTAF0	TCRF で設定したクロックバスでアップカウントを行います。	—	
ECNTBF0	外部入力の立ち上がりエッジの個数をカウントします。	0x0000	

2.13.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                : C/C++ compiler package for the SuperH RISC engine family
29 *                : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Active edge input interval.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long Period_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PHCR
63  b15:12 Reserved
64  b11:10 PH5MD[1:0] = 00 PH5 入出力 (ポート)
65  b9:8 PH4MD[1:0] = 00 PH4 入出力 (ポート)
66  b7:6 PH3MD[1:0] = 00 PH3 入出力 (ポート)
67  b5:4 PH2MD[1:0] = 00 PH2 入出力 (ポート)
68  b3:2 PH1MD[1:0] = 00 PH1 入出力 (ポート)
69  b1:0 PH0MD[1:0] = 11 TIF0A 入力 (ATU-III) */
70  PORTH.CR.WORD = 0x0003; /* PH0 を TIF0A(エッジ入力用端子)に設定 */
71
72  /* ATU-III(有効エッジ入力間隔計測)の設定 */
73  /* Configure TCRF0
74  b7:5 CKSELF0[2:0] = 000 クロックバス 0
75  b4:2 MDF0[2:0] = 001 有効エッジ入力間隔計測
76  b1:0 EGSELEn[1:0] = 01 立ち上がりエッジ */
77  ATUF.SUBBLOCK[0].TCRF.BYTE= 0x05; /* サブブロック 0 の時間計測カウンタのクロックソースを
    「クロックバス 0」,
78     動作モードを「有効エッジ入力間隔計測」,
79     計測エッジを「立上りエッジ」に それぞれ設定 */
80
81  /* Configure GRBF0
82  b0 GRBF0[15:0] アウトプットコンペア値 */
83  ATUF.SUBBLOCK[0].GRBF = 0x0005; /* 検出するエッジの回数を 5 回に設定 */
84
85  /* Configure ECNTBF0
86  b0 ECNTBF0[15:0] イベントカウンタ */
87  ATUF.SUBBLOCK[0].ECNTBF = 0x0000; /* イベントカウンタの初期値を 0 に設定 */
88
89  /* Configure TIERF0
90  b7:4 Reserved
91  b3 OVECF0 = 0 OVFCF0 による割り込みを禁止
92  b2 OVEBF0 = 0 OVFBF0 による割り込みを禁止
93  b1 OVEAF0 = 0 OVFAF0 による割り込みを禁止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94     b0 ICEF0 = 1 ICFF0 による割り込みを許可 */
95     ATUF.SUBBLOCK[0].TIERF.BYTE = 0x01; /* TIF0 のインプットキャプチャ割り込み(ICIF0)を許
96     可 */
97
98     /* Configure TSTRF
99     b0 STRF0 = 1 ECNTAF0、ECNTBF0、および ECNTCF0 のカウント動作を許可 */
100    ATUF.TSTRF.BIT.STRF0 = 1; /* サブブロック 0 のカウントをスタート */
101
102    /* Configure PSCR0
103    b15:10 Reserved
104    b9:0 PSC0 = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
105    ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
106
107    /* Configure IPR16
108    b15:12 ATU-F (ICIF0~ICIF3) = H'F 対応する割り込みの優先順位を 15 に設定
109    b11:8 ATU-F (ICIF4~ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
110    b7:4 ATU-F (ICIF8~ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
111    b3:0 ATU-F (ICIF12~ICIF15) = H'0 対応する割り込みの優先順位を 0 に設定 */
112    INTC.IPR16.WORD = 0xF000; /* ICIF0 の割り込み優先順位を 15 に設定 */
113
114    set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
115
116    /* Configure ATUENR
117    b15:10 Reserved
118    b9 TJE = 0 タイマ J のカウント動作を停止
119    b8 THE = 0 タイマ H のカウント動作を停止
120    b7 TGE = 0 タイマ G のカウント動作を停止
121    b6 TFE = 1 タイマ F のカウント動作を許可
122    b5 TEE = 0 タイマ E のカウント動作を停止
123    b4 TDE = 0 タイマ D のカウント動作を停止
124    b3 TCE = 0 タイマ C のカウント動作を停止
125    b2 TBE = 0 タイマ B のカウント動作を停止
126    b1 TAE = 0 タイマ A のカウント動作を停止
127    b0 PSCE = 1 プリスケーラのカウント動作を許可 */
128    ATUCTRL.ATUENR.WORD = 0x0041; /* タイマ F およびプリスケーラのカウントをスタート */
129
130    while(1);
131    } /* End of function main() */
132
133    /*****
134    * Function Name : ICIF0
135    * Description   : インプットキャプチャ割り込み
136    * Arguments    : none
137    * Return Value  : none
138    *****/
139    void ICIF0(void)
140    {
141        Period_cnt = ATUF.SUBBLOCK[0].CDRF; /* 計測した値を変数 Period _cnt に保存(GRAF0) */
142
143        /* Configure TSTRF0

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```
143    b7:4 Reserved
144    b3 OVFCF0 = 0 ECNTCFn にオーバーフローなし
145    b2 OVBF0 = 0 ECNTBFn にオーバーフロー/アンダフローなし
146    b1 OVFAF0 = 0 ECNTAFn にオーバーフローなし
147    b0 ICFF0 = 0 サブブロック Fn にインプットキャプチャの検出なし */
148    ATUF.SUBBLOCK[0].TSRF.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
149    } /* End of function ICIF0() */
```

2.13.6 SH7253 グループでご使用の場合の変更点

本文 2.13 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIF0A 端子が、SH7253 グループでは、PJ4 端子（SH7254R グループでは、PH0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.13.5 サンプルプログラムリストの 70 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PHCR を 0x0003 に設定 (変更前)

SH7253 グループ : PJCR2A を 0x0001 に設定 (変更後)

2.14 動作例 13 一定時間内エッジカウント[タイマ F]

2.14.1 概要

図 2.25に示すように、一定時間内での有効な入力エッジの回数を記録します。タイマ F の動作モードは、一定期間内エッジカウントに設定します。

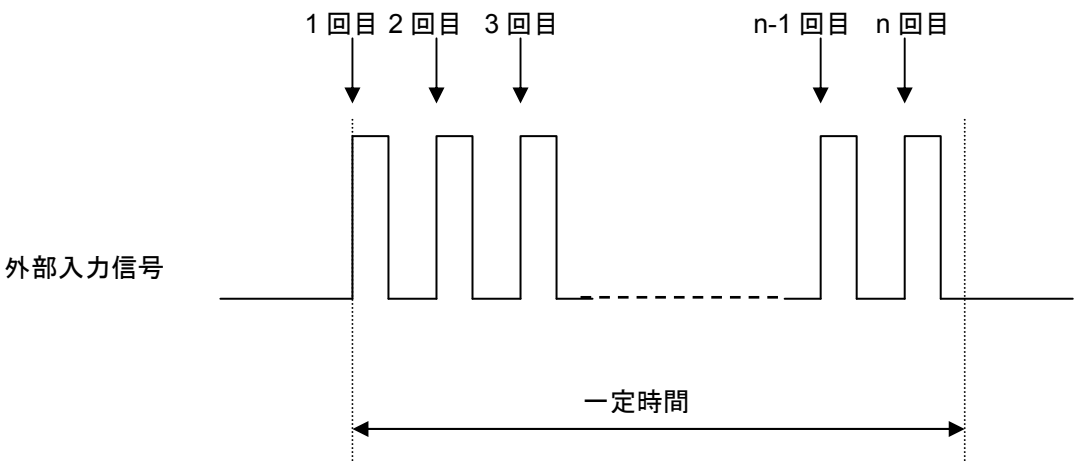


図 2.25 一定時間内エッジカウントの概要（立ち上がりエッジのカウント時）

2.14.2 使用機能説明

表 2.13に使用端子及び関連レジスタの機能割り付けを示します。

表 2.13 ATU-III機能割り付け

使用端子		機能
端子	TIF0A	外部から信号を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TIERF0	割り込みの許可をします。
	TCRF0	タイマ F 各サブブロックの動作を設定します。
	GRAF0	エッジを検出する期間を設定します。
	ECNTAF0	TCRF で設定したクロックバスでアップカウントを行います。
	ECNTBF0	有効な外部信号の回数を記録します。
	TSRF0	タイマ F サブブロック 0 の状態が格納されます。
	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
INTC レジスタ	IPR16	割り込みの優先順位を設定します。
PFC レジスタ	PHCR	端子機能を設定します。

2.14.3 動作説明

図 2.26に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって一定時間内の有効エッジ入力回数を計測します。

エッジ回数の計測期間は、時間計測カウンタレジスタで設定します。

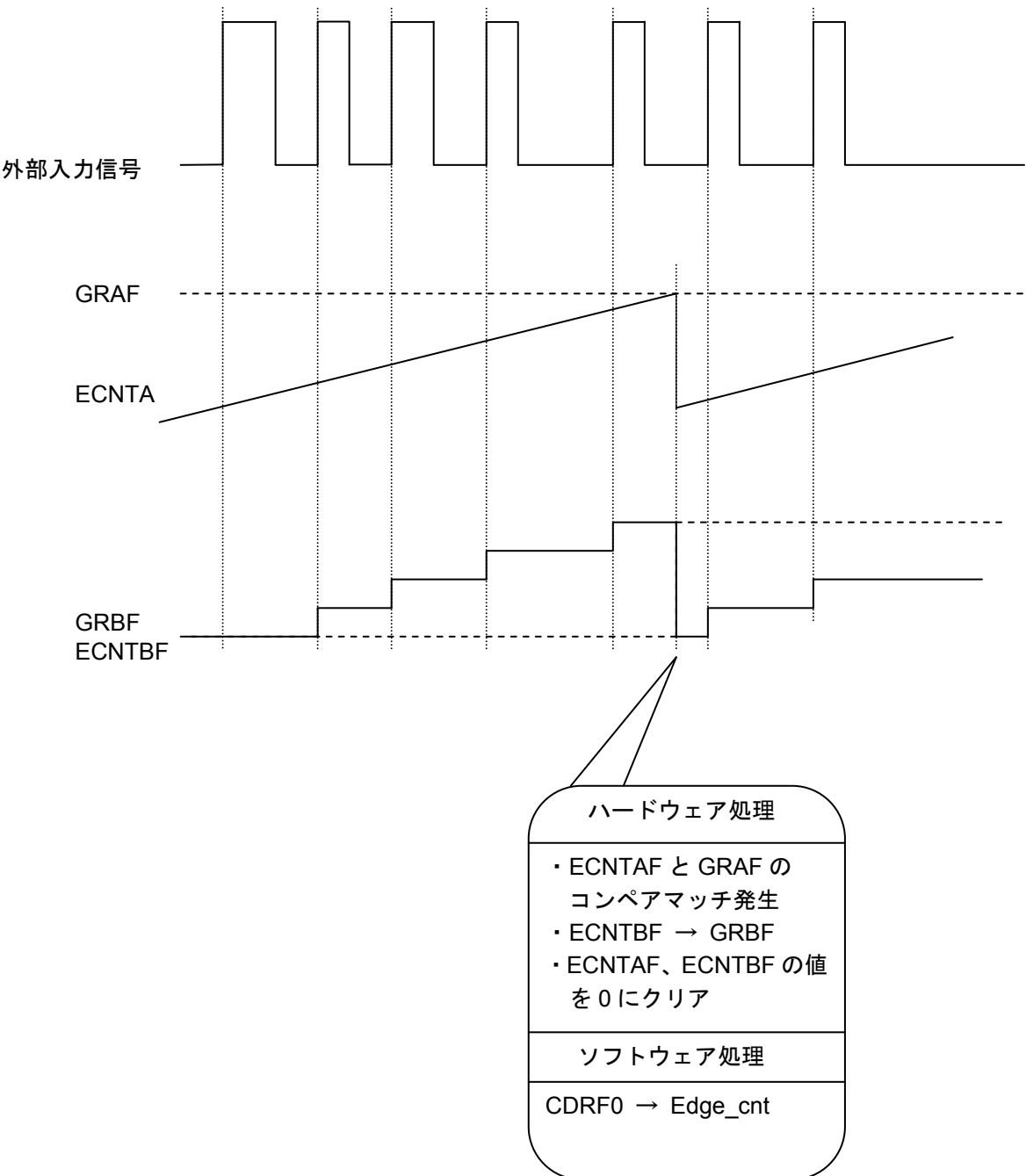


図 2.26 有効エッジ入力間隔計測動作原理

2.14.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-Ⅲの初期設定を行います。
計測値の保存ルーチン	ICIF0	計測した値を変数に保存します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
Edge_cnt	計測したエッジ数を保存します	unsigned long	計測値の保存ルーチン

● 使用レジスタの設定

レジスタ名	機能	設定値	使用モジュール名
PHCR	PH0 端子を TIF0A に設定します。	0x0003	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-Ⅲのタイマ F およびプリスケアラのカウンタ開始を許可します。	0x0041	
IPR16	割り込みの優先順位を設定します。	0xF000	
TIERF0	割り込みの許可をします。	0x01	
TSTRF	タイマ F サブブロック 0 のカウンタをスタートさせる。	0x00000001	
TCRF0	サブブロック 0 を「クロックバス 0 でカウンタ」、「一定期間内エッジカウンタ」「計測エッジは立ち上がり」に設定します。	0x01	
GRAF0	エッジを計測する期間を 0x50 カウンタに設定します。	0x00005000	計測値の保存ルーチン
TSRF0	タイマ F のインプットキャプチャフラグをクリアします。	0x00	

2.14.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Edge counts within a certain period .
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39 /*****
40 Includes <System Includes> , "Project Includes"
41 *****/
42 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
43 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
44
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long Edge_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments    : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PHCR
63  b15:12 Reserved
64  b11:10 PH5MD[1:0] = 00 PH5 入出力 (ポート)
65  b9:8 PH4MD[1:0] = 00 PH4 入出力 (ポート)
66  b7:6 PH3MD[1:0] = 00 PH3 入出力 (ポート)
67  b5:4 PH2MD[1:0] = 00 PH2 入出力 (ポート)
68  b3:2 PH1MD[1:0] = 00 PH1 入出力 (ポート)
69  b1:0 PH0MD[1:0] = 11 TIF0A 入力 (ATU-III) */
70  PORTH.CR.WORD = 0x0003; /* PH0 を TIF0A(エッジ入力用端子)に設定 */
71
72  /* ATU-III(一定期間内エッジカウント)の設定 */
73  /* Configure TCRF0
74  b7:5 CKSELF0[2:0] = 000 クロックバス 0
75  b4:2 MDF0[2:0] = 000 一定期間内エッジカウント
76  b1:0 EGSELEn[1:0] = 01 立ち上がりエッジ */
77  ATUF.SUBBLOCK[0].TCRF.BYTE= 0x01; /* サブブロック 0 の時間計測カウンタの設定: クロック
    バス 0
78  動作モード: B'000 = 一定期間内エッジカウント
79  計測エッジ: B'01 = 立上りエッジ */
80
81  /* Configure GRAF0
82  b31:8 GRAF0[23:0] 時間計測カウンタ A のインプットキャプチャ値
83  b7:0 Reserved */
84  ATUF.SUBBLOCK[0].GRAF = 0x00005000; /* エッジ計測期間を 0x50 カウントに設定 */
85
86  /* Configure TIERF0
87  b7:4 Reserved
88  b3 OVECF0 = 0 OVFCF0 による割り込みを禁止
89  b2 OVEBF0 = 0 OVFBF0 による割り込みを禁止
90  b1 OVEAF0 = 0 OVFAF0 による割り込みを禁止
91  b0 ICEF0 = 1 ICFF0 による割り込みを許可 */
92  ATUF.SUBBLOCK[0].TIERF.BYTE = 0x01; /* ICEF0 セットによる割り込み許可 */
93
94  /* Configure TSTRF

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

95     b0 STRF0 = 1 ECNTAF0、ECNTBF0、および ECNTCF0 のカウント動作を許可 */
96     ATUF.TSTRF.BIT.STRF0 = 1; /* サブブロック 0 のカウントをスタート */
97
98     /* Configure PSCR0
99     b15:10 Reserved
100    b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
101    ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
102
103    /* Configure IPR16
104    b15:12 ATU-F (ICIF0~ICIF3) = H'F 対応する割り込みの優先順位を 15 に設定
105    b11:8 ATU-F (ICIF4~ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
106    b7:4 ATU-F (ICIF8~ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
107    b3:0 ATU-F (ICIF12~ICIF15) = H'0 対応する割り込みの優先順位を 0 に設定 */
108    INTC.IPR16.WORD = 0xF000; /* ICIF0 の割り込み優先順位を 15 に設定 */
109
110    set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
111
112    /* Configure ATUENR
113    b15:10 Reserved
114    b9 TJE = 0 タイマ J のカウント動作を停止
115    b8 THE = 0 タイマ H のカウント動作を停止
116    b7 TGE = 0 タイマ G のカウント動作を停止
117    b6 TFE = 1 タイマ F のカウント動作を許可
118    b5 TEE = 0 タイマ E のカウント動作を停止
119    b4 TDE = 0 タイマ D のカウント動作を停止
120    b3 TCE = 0 タイマ C のカウント動作を停止
121    b2 TBE = 0 タイマ B のカウント動作を停止
122    b1 TAE = 0 タイマ A のカウント動作を停止
123    b0 PSCE = 1 プリスケーラのカウント動作を許可 */
124    ATUCTRL.ATUENR.WORD = 0x0041; /* タイマ F およびプリスケーラのカウントをスタート*/
125
126    while(1);
127 } /* End of function main() */
128
129 /******
130 * Function Name : ICIF0
131 * Description    : インพุットキャプチャ割り込み
132 * Arguments      : none
133 * Return Value   : none
134 *****/
135 void ICIF0(void)
136 {
137     /* Configure CDRF0
138     b31:8 CDRFn[23:0] GRBF0 のデータ
139     b7:0 Reserved */
140     Edge_cnt = ATUF.SUBBLOCK[0].CDRF; /* 計測した値を変数 Edge_cnt に保存 (GRBF0) */
141
142     /* Configure TSTRF0
143     b7:4 Reserved
144     b3 OVFCF0 = 0 オーバフローフラグをクリア

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```
145    b2 OVBF0 = 0 オーバフロー／アンダフローフラグをクリア
146    b1 OVFAF0 = 0 オーバフローフラグをクリア
147    b0 ICFF0 = 0 インพุットキャプチャの検出フラグをクリア */
148    ATUF.SUBBLOCK[0].TSRF.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
149    } /* End of function ICIF0() */
```

2.14.6 SH7253 グループでご使用の場合の変更点

本文 2.14 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIF0A 端子が、SH7253 グループでは、PJ4 端子（SH7254R グループでは、PH0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.14.5 サンプルプログラムリストの 70 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PHCR を 0x0003 に設定（変更前）

SH7253 グループ : PJCR2A を 0x0001 に設定（変更後）

2.15 動作例 14 入力 High/Low 期間計測[タイマ F]

2.15.1 概要

図 2.27に示すように、入力パルス内規定回数内での High（または Low）レベル出力の時間を記録します。本動作例では、タイマ F の動作モードは入力 High/Low 期間計測モードに設定し、入力パルスの High 期間を計測対象とします。

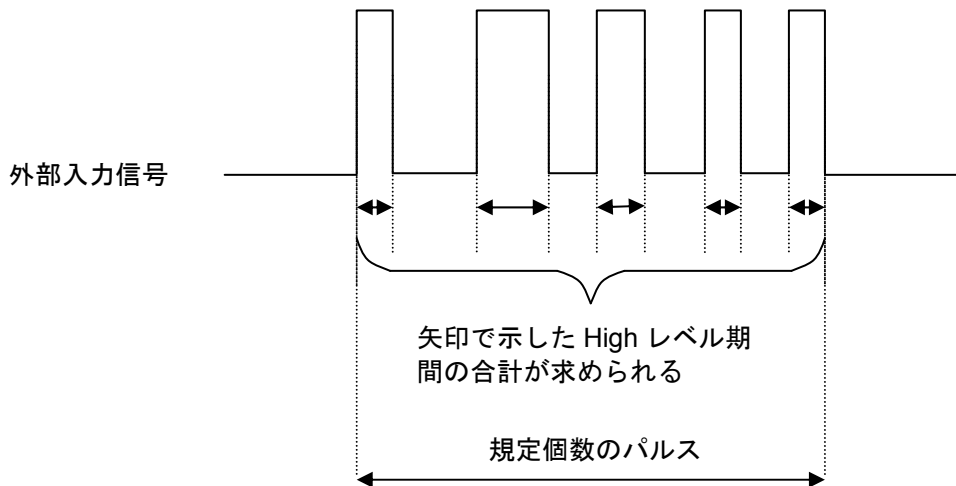


図 2.27 High/Low 期間計測の概要

2.15.2 使用機能説明

表 2.14に使用端子及び関連レジスタの機能割り付けを示します。

表 2.14 ATU-III機能割り付け

使用端子		機能
端子	TIF0A	外部から信号を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TIERF0	割り込みの許可をします。
	TCRF0	タイマ F 各サブブロックの動作を設定します。
	GRBF0	検出するパルスの個数を設定します。
	ECNTAF0	設定したレベルが入力されている期間をカウントします。
	ECNTBF0	有効な外部信号の回数を記録します。
	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
INTC レジスタ	IPR16	割り込み優先順位を設定します。

PFC レジスタ	PHCR	端子機能を設定します。
----------	------	-------------

2.15.3 動作説明

図 2.28に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって一定時間内の有効エッジ入力の回数を計測します。入力ハイ/ローレベル期間計測モードでは、検出するパルスの回数を設定し、設定回数検出完了後にインプットキャプチャ割り込みを行います。この時のカウンタの値を用いることで期間計測します。

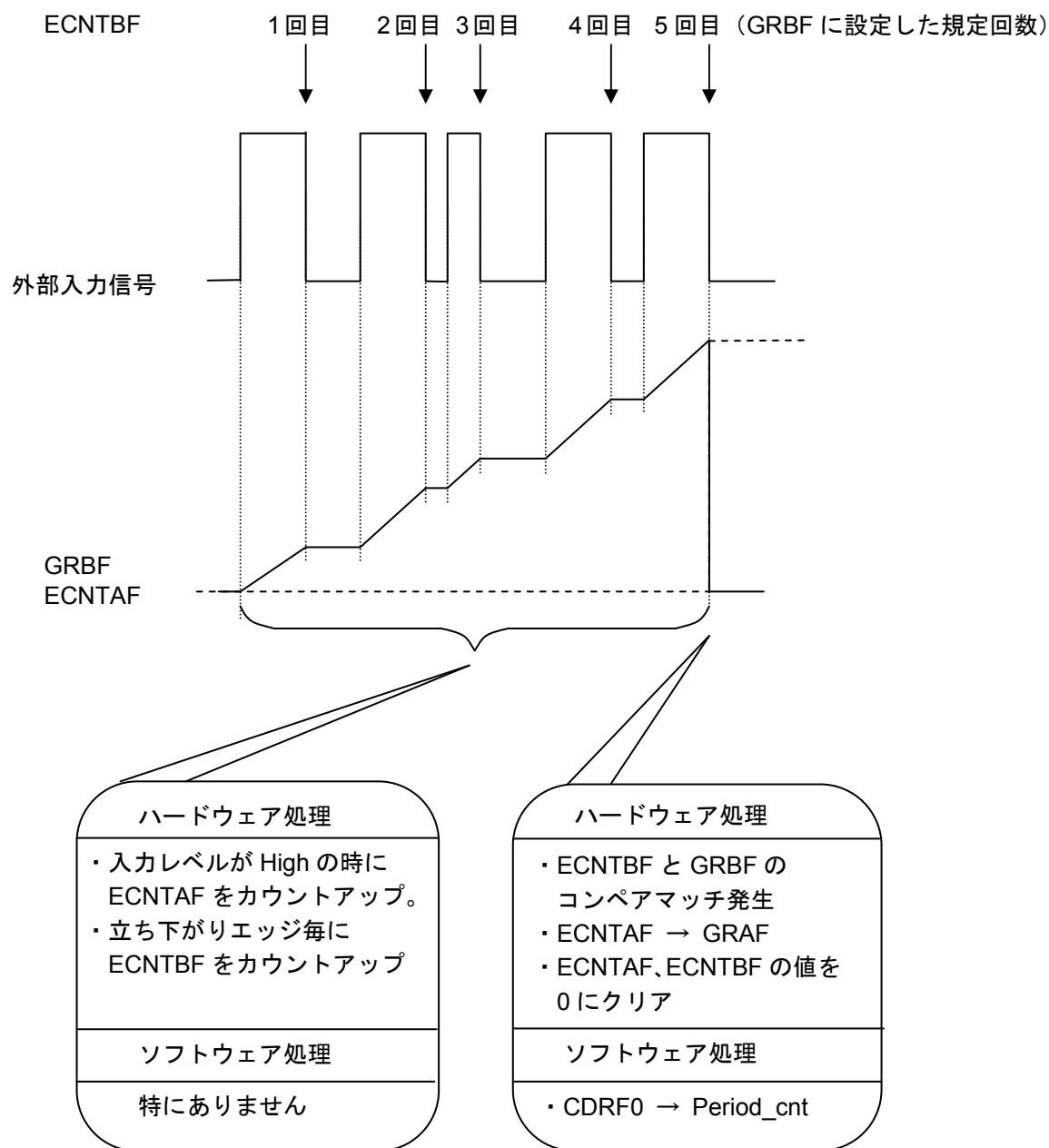


図 2.28 有効エッジ入力間隔計測動作原理

2.15.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
計測値の保存ルーチン	ICIF0	ICIF0 セットで割り込みを発生し、計測した値を変数で保存します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
Period_cnt	計測した High(または Low)期間の合計値を保存します	unsigned long	計測値の保存ルーチン

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PHCR	PH0 端子を TIF0A に設定します。	0x0003	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
IPR16	割り込みの優先順位を設定します。	0xF000	
TIERF0	割り込みの許可をします。	0x01	
ATUENR	ATU-IIIのタイマ F およびプリスケアラのカウント開始を許可します。	0x0041	
TSTRF	タイマ F サブブロック 0 のカウント動作を許可します。	STRF0 = 1	
TCRF0	サブブロック 0 を「クロックバス 0 でカウント」、「入力 High/Low 期間計測」「計測期間は High」に設定します。	0x09	
GRBF0	計測パルスの回数を 5 回に設定します。	0x0005	
TSRF0	タイマ F サブブロック 0 のインプットキャプチャフラグをクリアします。	0x00	

2.15.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Measurement of High / Low Input period.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long Period_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PHCR
63  b15:12 Reserved
64  b11:10 PH5MD[1:0] = 00 PH5 入出力 (ポート)
65  b9:8 PH4MD[1:0] = 00 PH4 入出力 (ポート)
66  b7:6 PH3MD[1:0] = 00 PH3 入出力 (ポート)
67  b5:4 PH2MD[1:0] = 00 PH2 入出力 (ポート)
68  b3:2 PH1MD[1:0] = 00 PH1 入出力 (ポート)
69  b1:0 PH0MD[1:0] = 11 TIF0A 入力 (ATU-III) */
70  PORTH.CR.WORD = 0x0003; /* PH0 を TIF0A に設定 (外部信号の入力端子) */
71
72  /* ATU-III (入力ハイ/ロー期間計測) の設定 */
73  /* Configure TCRF0
74  b7:5 CKSELF0[2:0] = 000 クロックバス 0
75  b4:2 MDF0[2:0] = 010 入力ハイ/ローレベル期間計測
76  b1:0 EGSELFn[1:0] = 01 立ち上がりエッジ */
77  ATUF.SUBBLOCK[0].TCRF.BYTE=0x09; /* 時間計測カウンタのクロックソースを「クロックバス 0」,
78  サブブロック 0 の動作モードを「入力ハイ/ロー期間計測」,
79  計測出力を「ハイ期間計測」に それぞれ設定 */
80
81  /* Configure GRBF0
82  b15:0 GRBF0[15:0] イベントカウンタのインプットキャプチャ値 */
83  ATUF.SUBBLOCK[0].GRBF = 0x0005; /* 検出するパルスの回数を 5 回に設定 */
84
85  /* Configure TIERF0
86  b7:4 Reserved
87  b3 OVECF0 = 0 OVFCF0 による割り込みを禁止
88  b2 OVEBF0 = 0 OVFBF0 による割り込みを禁止
89  b1 OVEAF0 = 0 OVFAF0 による割り込みを禁止
90  b0 ICEF0 = 1 ICFF0 による割り込みを許可 */
91  ATUF.SUBBLOCK[0].TIERF.BYTE = 0x01; /* ICEF0 セットによる割り込みを許可 */
92
93  /* Configure TSTRF
94  b0 STRF0 = 1 ECNTAF0、ECNTBF0、および ECNTCF0 のカウント動作を許可 */

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

95  ATUF.TSTRF.BIT.STRF0 = 1; /* サブブロック 0 のカウントをスタート */
96
97  /* Configure PSCR0
98  b15:10 Reserved
99  b9:0 PSC0 = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
100 ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
101
102 /* Configure IPR16
103 b15:12 ATU-F (ICIF0~ICIF3) = H'F 対応する割り込みの優先順位を 15 に設定
104 b11:8 ATU-F (ICIF4~ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
105 b7:4 ATU-F (ICIF8~ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
106 b3:0 ATU-F (ICIF12~ICIF15) = H'0 対応する割り込みの優先順位を 0 に設定 */
107 INTC.IPR16.WORD = 0xF000; /* ICIF0 の割り込み優先順位を 15 に設定 */
108
109 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
110
111 /* Configure ATUENR
112 b15:10 Reserved
113 b9 TJE = 0 タイマ J のカウント動作を停止
114 b8 THE = 0 タイマ H のカウント動作を停止
115 b7 TGE = 0 タイマ G のカウント動作を停止
116 b6 TFE = 1 タイマ F のカウント動作を許可
117 b5 TEE = 0 タイマ E のカウント動作を停止
118 b4 TDE = 0 タイマ D のカウント動作を停止
119 b3 TCE = 0 タイマ C のカウント動作を停止
120 b2 TBE = 0 タイマ B のカウント動作を停止
121 b1 TAE = 0 タイマ A のカウント動作を停止
122 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
123 ATUCTRL.ATUENR.WORD = 0x0041; /* タイマ F およびプリスケーラのカウントをスタート */
124
125 while(1);
126 } /* End of function main() */
127
128 /*****
129 * Function Name : ICIF0
130 * Description   : インพุットキャプチャ割り込み
131 * Arguments     : none
132 * Return Value  : none
133 *****/
134 void ICIF0(void)
135 {
136  /* Configure CDRF0
137  b31:8 CDRFn[23:0] GRAF0 のデータ
138  b7:0 Reserved */
139  Period_cnt = ATUF.SUBBLOCK[0].CDRF; /* 変数 Period_cnt に計測したエッジ数を保存 */
140
141  /* Configure TSTRF0
142  b7:4 Reserved
143  b3 OVFCF0 = 0 オーバフローフラグをクリア
144  b2 OVFBF0 = 0 オーバフロー/アンダフローフラグをクリア

```



```
145    b1 OVFAF0 = 0 オーバフローフラグをクリア
146    b0 ICFF0 = 0 インพุットキャプチャの検出フラグをクリア */
147    ATUF.SUBBLOCK[0].TSRF.BYTE &= 0x00; /* ICEF0 を 0 にクリア */
148    } /* End of function ICIF0() */
```

2.15.6 SH7253 グループでご使用の場合の変更点

本文 2.15 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIF0A 端子が、SH7253 グループでは、PJ4 端子（SH7254R グループでは、PH0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.15.5 サンプルプログラムリストの 70 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PHCR を 0x0003 に設定 (変更前)

SH7253 グループ : PJCR2A を 0x0001 に設定 (変更後)

2.16 動作例 15 PWM 入力波形計測[タイマ F]

2.16.1 概要

図 2.29に示すように、入力される一定回数の PWM 波形についてオフデューティと PWM サイクルを計測します。タイマ F の動作モードは、PWM 入力波形計測モードに設定します。

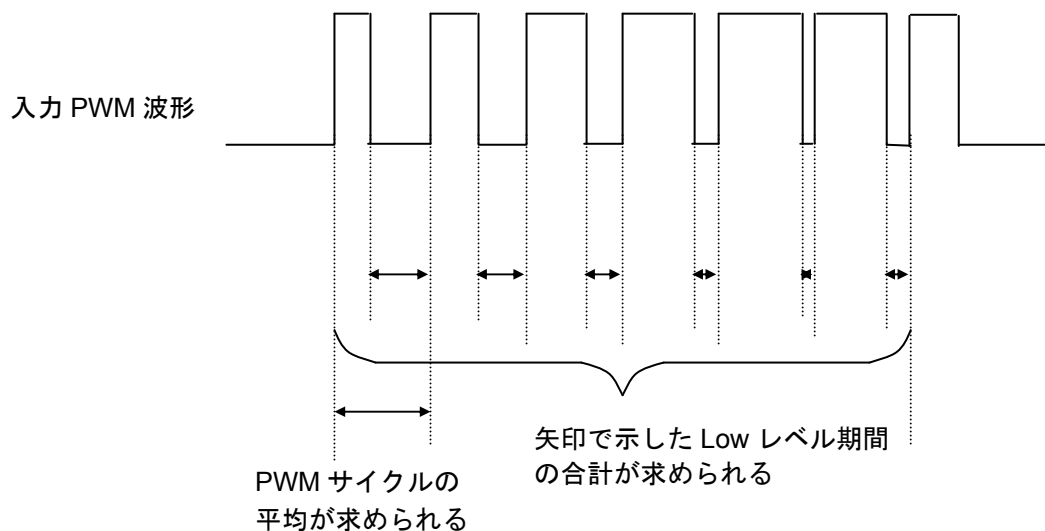


図 2.29 PWM 入力波形計測の概要

2.16.2 使用機能説明

表 2.15に使用端子及び関連レジスタの機能割り付けを示します。

表 2.15 ATU-III機能割り付け

使用端子		機能
端子	TIF0A	外部から信号を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TCRF0	タイマ F 各サブブロックの動作を設定します。
	GRBF0	検出するパルスの個数を設定します。
	ECNTAF0	設定したレベルが入力されている期間をカウントします。
	ECNTBF0	有効な外部信号の回数を記録します。
	ECNTCF0	外部入力のエッジの個数をカウントします。
	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
	TSRF0	タイマ F サブブロック 0 の状態が格納されます。
	CDRF0	タイマ F キャプチャ出力レジスタです。
	GRCF0	PWM 波形のサイクルの値を保存
	TIERF0	タイマステータスレジスタ F (TSRF) のステータスフラグに対応する割り込みを許可するか禁止するか指定します。
INTC レジスタ	IPR16	割り込み優先順位を設定します。
PFC レジスタ	PHCR	端子機能を設定します。

2.16.3 動作説明

図 2.30に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって PWM 波形のオフデューティと、PWM サイクルを計測します。

設定された数の入力 PWM 波形を検出するとインプットキャプチャ割り込みが発生し、計測開始からのカウンタ値と入力 PWM 波形 OFF 時のみ更新されたカウンタ値を得ます。

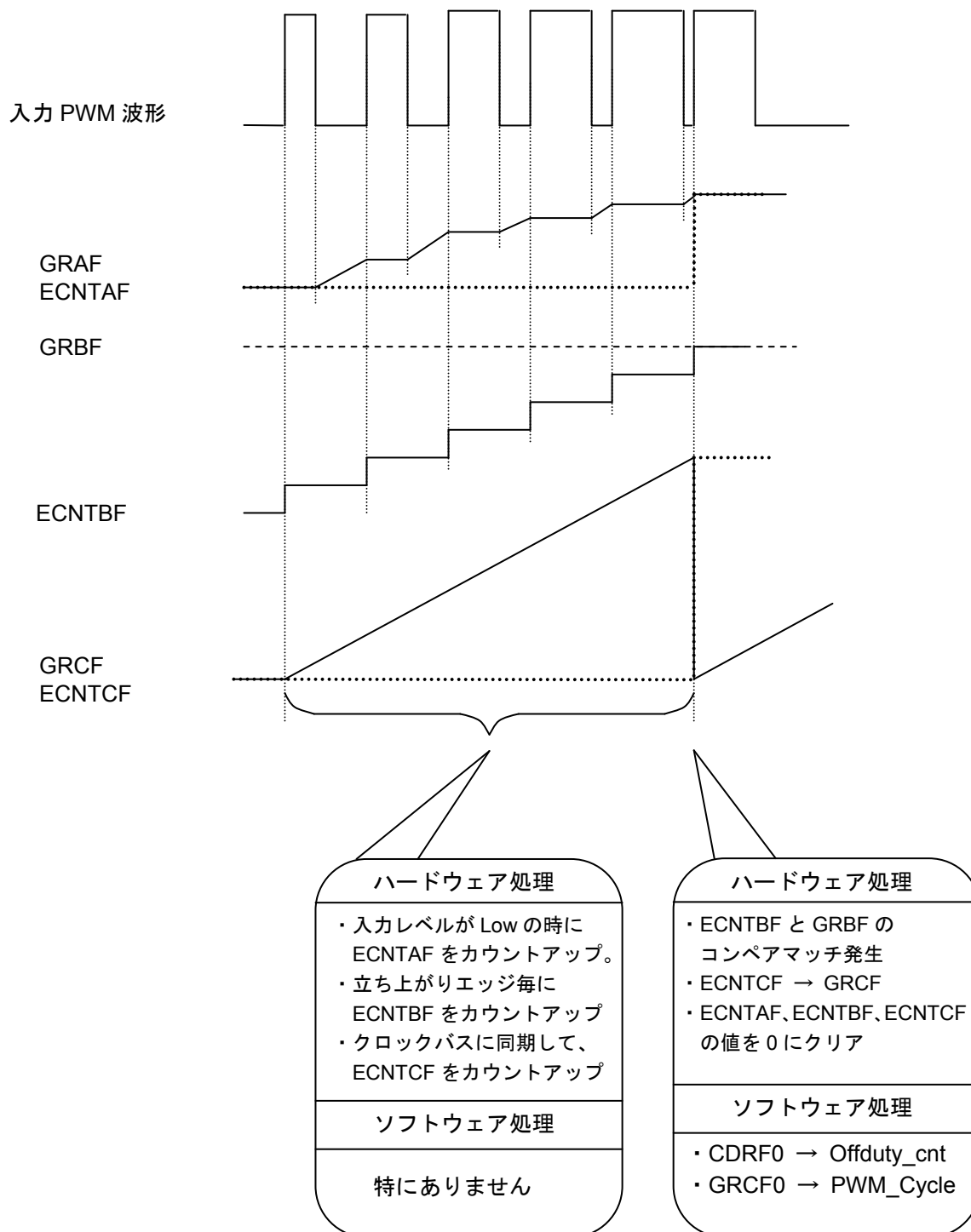


図 2.30 有効エッジ入力間隔計測動作原理

2.16.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
計測値の保存ルーチン	ICIF0	計測した値を変数に保存します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
Offduty_cnt	計測期間内のオフデューティの期間の合計値を保存します。	unsigned	計測値の保存ルーチン
PWM_Cycle	直前に入力された PWM 波のサイクルを保存します。	long	

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PHCR	PH0 端子を TIF0A に設定します。	0x0003	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIのタイマ F およびプリスケアラのカウント開始を許可します。	0x0061	
TSTRF	タイマ F サブブロック 0 のカウントをスタートさせます。	0x00000001	
TCRF0	サブブロック 0 を「クロックバス 0 でカウント」、「PWM 入力波形計測」「計測エッジは立ち上がり」に設定します。	0x11	
GRBF0	計測パルスの回数を 6 回に設定します。	0x0006	
TIERF0	TIF0 のインプットキャプチャ割り込み(ICIF0)を許可します。	0x01	
CDRF0	計測したオフデューティ合計を保存します。	—	計測値の保存ルーチン
GRCF0	直前の PWM サイクルの値を保存します。	—	
TSRF0	タイマ F サブブロック 0 のステータスレジスタをクリアします。	0x00	

2.16.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Measurement of PWM input waveform.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long Offduty_cnt, PWM_Cycle;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58  void main(void)
59  {
60  /* ポートの設定 */
61  /* Configure PHCR
62  b15:12 Reserved
63  b11:10 PH5MD[1:0] = 00 PH5 入出力 (ポート)
64  b9:8 PH4MD[1:0] = 00 PH4 入出力 (ポート)
65  b7:6 PH3MD[1:0] = 00 PH3 入出力 (ポート)
66  b5:4 PH2MD[1:0] = 00 PH2 入出力 (ポート)
67  b3:2 PH1MD[1:0] = 00 PH1 入出力 (ポート)
68  b1:0 PH0MD[1:0] = 11 TIF0A 入力 (ATU-III) */
69  PORTH.CR.WORD = 0x0003; /* PH0 を TIF0A(エッジ入力用端子)に設定 */
70
71  /* ATU-III(PWM 入力波形計測)の設定 */
72  /* Configure TCRF0
73  b7:5 CKSELF0[2:0] = 000 クロックバス 0
74  b4:2 MDF0[2:0] = 100 PWM 入力波形計測
75  b1:0 EGSELEn[1:0] = 01 立ち上がりエッジ */
76  ATUF.SUBBLOCK[0].TCRF.BYTE= 0x11; /* サブブロック 0 の時間計測カウンタのクロックソースを
    「クロックバス 0」,
77  動作モードを「PWM 入力波形計測」,
78  計測エッジを「立上りエッジ」に それぞれ設定 */
79
80  /* Configure GRBF0
81  b15:0 GRBF0[15:0] イベントカウンタのインプットキャプチャ値 */
82  ATUF.SUBBLOCK[0].GRBF = 0x0006; /* 検出するエッジの回数を 6 周期分に設定 */
83
84  /* Configure TIERF0
85  b7:4 Reserved
86  b3 OVECF0 = 0 OVFCF0 による割り込みを禁止
87  b2 OVEBF0 = 0 OVFBF0 による割り込みを禁止
88  b1 OVEAF0 = 0 OVFAF0 による割り込みを禁止
89  b0 ICEF0 = 1 ICFF0 による割り込みを許可 */
90  ATUF.SUBBLOCK[0].TIERF.BYTE = 0x01; /* TIF0 のインプットキャプチャ割り込み(ICIF0)を
    許可 */
91
92  /* Configure TSTRF
93  b0 STRF0 = 1 ECNTAF0、ECNTBF0、および ECNTCF0 のカウント動作を許可 */

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94  ATUF.TSTRF.BIT.STRF0 = 1; /* サブブロック 0 のカウントをスタート */
95
96  /* Configure PSCR0
97  b15:10 Reserved
98  b9:0 PSC0 = H'9 プリスケアラの分周比を 10(設定値は-1)に設定 */
99  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケアラの分周比を 10 に設定 */
100
101  /* Configure IPR16
102  b15:12 ATU-F (ICIF0~ICIF3) = H'F 対応する割り込みの優先順位を 15 に設定
103  b11:8 ATU-F (ICIF4~ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
104  b7:4 ATU-F (ICIF8~ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
105  b3:0 ATU-F (ICIF12~ICIF15) = H'0 対応する割り込みの優先順位を 0 に設定 */
106  INTC.IPR16.WORD = 0xF000; /* ICIF0 の割り込み優先順位を 15 に設定 */
107
108  set_imask(0); /* 割り込みマスケレベルを 0 に設定 */
109
110  /* Configure ATUENR
111  b15:10 Reserved
112  b9 TJE = 0 タイマ J のカウント動作を停止
113  b8 THE = 0 タイマ H のカウント動作を停止
114  b7 TGE = 0 タイマ G のカウント動作を停止
115  b6 TFE = 1 タイマ F のカウント動作を許可
116  b5 TEE = 1 タイマ E のカウント動作を許可
117  b4 TDE = 0 タイマ D のカウント動作を停止
118  b3 TCE = 0 タイマ C のカウント動作を停止
119  b2 TBE = 0 タイマ B のカウント動作を停止
120  b1 TAE = 0 タイマ A のカウント動作を停止
121  b0 PSCE = 1 プリスケアラのカウント動作を許可 */
122  ATUCTRL.ATUENR.WORD = 0x0061; /* タイマ E, F およびプリスケアラのカウントをスタート */
123
124  while(1);
125  } /* End of function main() */
126
127  /*****
128  * Function Name : ICIF0
129  * Description   : インプットキャプチャ割り込み
130  * Arguments    : none
131  * Return Value  : none
132  *****/
133  void ICIF0(void)
134  {
135  /* Configure CDRF0
136  b31:8 CDRF0[23:0] GRAF0 のデータ
137  b7:0 Reserved */
138  Offduty_cnt = ATUF.SUBBLOCK[0].CDRF; /* 変数 Edge_cnt に計測したオフデューティ合計を保存 */
139
140  /* Configure GRCF0
141  b31:8 GRCF0[23:0] ECNTCF0 のデータ
142  b7:0 Reserved */

```

```
143    PWM_Cycle = ATUF.SUBBLOCK[0].GRCF; /* 変数 PWM_Cycle に直前の PWM サイクルの値を保存 */
144
145    /* Configure TSRF0
146    b7:4 Reserved
147    b3 OVFCF0 = 0 オーバフローフラグをクリア
148    b2 OVFBF0 = 0 オーバフロー/アンダフローフラグをクリア
149    b1 OVFAF0 = 0 オーバフローフラグをクリア
150    b0 ICFF0 = 0 インพุットキャプチャの検出フラグをクリア */
151    ATUF.SUBBLOCK[0].TSRF.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
152 } /* End of function ICIF0() */
```

2.16.6 SH7253 グループでご使用の場合の変更点

本文 2.16 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIF0A 端子が、SH7253 グループでは、PJ4 端子（SH7254R グループでは、PH0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.16.5 サンプルプログラムリストの 69 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PHCR を 0x0003 に設定 (変更前)

SH7253 グループ : PJCR2A を 0x0001 に設定 (変更後)

2.17 動作例 16 回転速度/パルス計測[タイマ F]

2.17.1 概要

図 2.31に示すように、入力されたエッジ数とエッジ入力時間を計測し、さらに直前の入力エッジとの間に現れる PWM 波形のオフデューティと PWM サイクルを測定します。動作モードは回転速度/パルス計測モードに設定します。

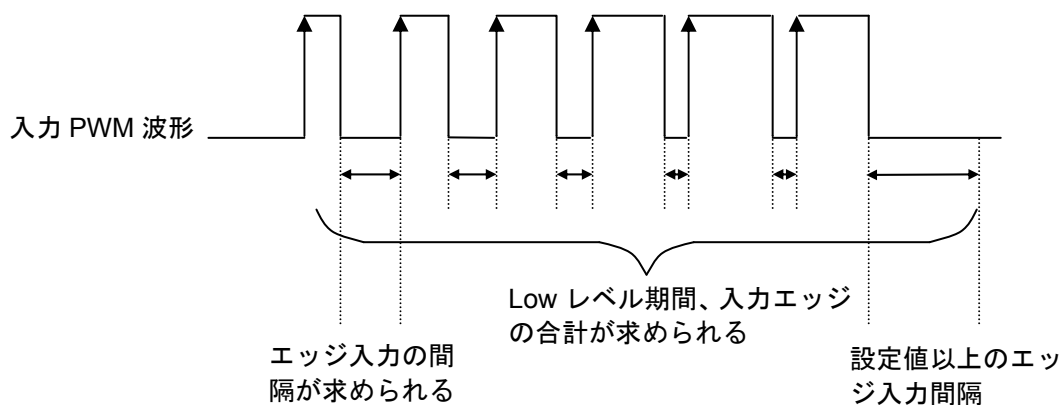


図 2.31 回転速度/パルス計測の概要

2.17.2 使用機能説明

表 2.16に使用端子及び関連レジスタの機能割り付けを示します。

表 2.16 ATU-III機能割り付け

使用端子		機能
端子	TIF12	外部から信号を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TCRF12	タイマ F 各サブブロックの動作を設定します。
	ECNTAF12	設定したレベルが入力されている期間をカウントします。
	ECNTBF12	有効な外部信号の回数を記録します。
	ECNTCF12	エッジの間隔をカウントします。
	GRAF12	ECNTAF12 の値をキャプチャします。
	GRBF12	エッジの最大間隔を設定します。
	GRCF12	ECNTCF12 の値をキャプチャします。
	GRDF12	動作開始からのカウント数をキャプチャします。
	CDRF12	ECNTBF12 の値をキャプチャします。
	TSRF12	タイマ F サブブロック 12 の状態が格納されます。
	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
PFC レジスタ	PHCR	端子機能を設定します。
INTC レジスタ	IPR16	TIF12 のインプットキャプチャ割り込みの優先度を設定します。

2.17.3 動作説明

図 2.32に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって PWM 波形のオフデューティ、PWM サイクルを計測します。下記では、カウンタ値 ECNTCF がオーバーフローフラグ設定値である GRBF にセットされるとインプットキャプチャ割り込みが発生し、計測に必要なカウンタ値を得ます。

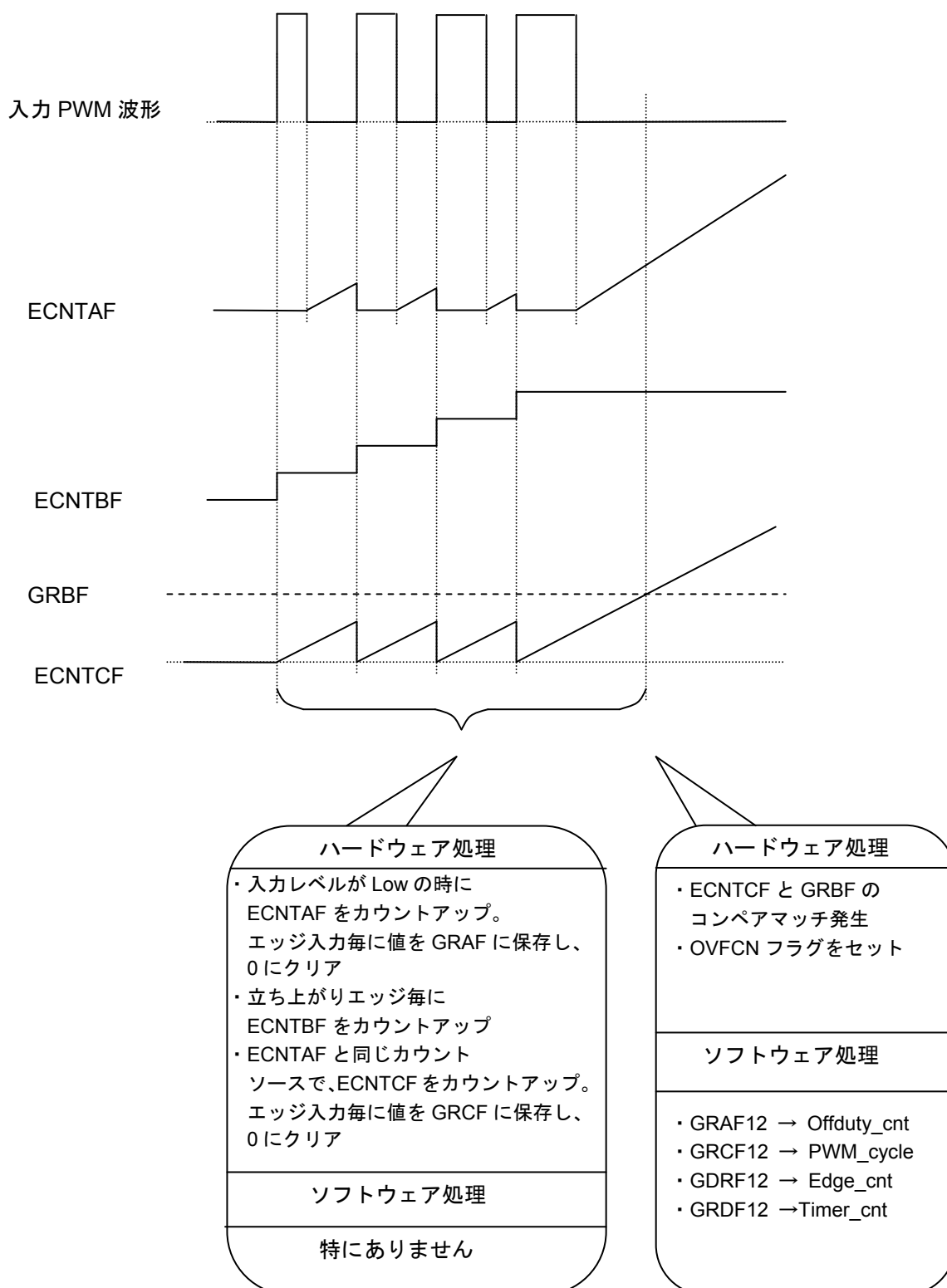


図 2.32 有効エッジ入力間隔計測動作原理

2.17.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
計測値の保存ルーチン	ICIF12	計測した値を変数に保存します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
Edge_cnt	動作開始からその時点までの入力されたエッジ数を保存します。	unsigned long	計測値の保存ルーチン
Timer_cnt	動作開始からその時点までの設定したクロックバスでのカウント数を保存します。		
Offduty_cnt	直前に入力された PWM 波形のオフデューティの値を保存します。		
PWM_Cycle	直前に入力された PWM 波のサイクルを保存します。		

● 使用レジスタの設定

レジスタ名	機能	設定値	使用モジュール名
PGCR2	PG14 端子を TIF12 に設定します。	0x3000	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIのタイマ F およびプリスケアラのカウント開始を許可します。	0x0061	
TSTRF	タイマ F サブブロック 12 のカウントをスタートさせます。	STRF12 = 1	
TCRF12	サブブロック 12 を「クロックバス 0 でカウント」、「回転速度/パルス計測」「計測エッジは立ち上がり」に設定します。	0x15	
GRBF12	オーバーフローを発生させるエッジ間隔を 0x110 に設定します。	0x0110	
TIERF12	ICFF12 のセットによるインプットキャプチャ割り込みを許可します。	0x0001	
IPR16	ICIF12 の割り込み優先度を 14 に設定します。	0x000E	

2.17.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Measurement of rotation speed and pulse .
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39 /*****
40 Includes <System Includes> , "Project Includes"
41 *****/
42 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
43 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
44
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long Offduty_cnt, PWM_cycle, Edge_cnt, Timer_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  Offduty_cnt = 0;
62  PWM_cycle   = 0;
63  Edge_cnt    = 0;
64  Timer_cnt   = 0;
65
66  /* ポートの設定 */
67  /* Configure PGCR2
68  b15:14 PG15MD[1:0] = 00 PG15 入出力 (ポート)
69  b13:12 PG14MD[1:0] = 11 TIF12 入力 (ATU-III)
70  b11:10 PG13MD[1:0] = 00 PG13 入出力 (ポート)
71  b9:8  PG12MD[1:0] = 00 PG12 入出力 (ポート)
72  b7:6  PG11MD[1:0] = 00 PG11 入出力 (ポート)
73  b5:4  PG10MD[1:0] = 00 PG10 入出力 (ポート)
74  b3:2  PG9MD[1:0]  = 00 PG9 入出力 (ポート)
75  b1:0  PG8MD[1:0]  = 00 PG8 入出力 (ポート) */
76  PORTG.CR2.WORD = 0x3000; /* PG14 を TIF12(エッジ入力用端子)に設定 */
77
78  /* ATU-III (回転速度/パルス計測)の設定 */
79  /* Configure TCRF12
80  b7:5  CKSELF12[2:0] = 000 クロックバス 0
81  b4:2  MDF12[2:0]   = 101 回転速度/パルス計測
82  b1:0  EGSELF12[1:0] = 01 立ち上がりエッジ */
83  ATUF.SUBBLOCK[12].TCRF.BYTE= 0x15; /* サブブロック 12 の時間計測カウンタのクロックソースを「ク
      ロックバス 0」,
84  動作モードを「回転速度/パルス計測」,
85  計測エッジを「立上りエッジ」に それぞれ設定 */
86
87  /* Configure GRBF0
88  b15:0 GRBF0[15:0] イベントカウンタのインプットキャプチャ値 */
89  ATUF.SUBBLOCK[12].GRBF = 0x0110; /* オーバフローフラグをセットするエッジ間隔を 0x0110 に
      設定 */
90
91  /* Configure TSRF12
92  b7:4 Reserved

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

93  b3 OVFCF12 = 0 ECNTCF12 にオーバーフローなし
94  b2 OVFBF12 = 0 ECNTBF12 にオーバーフロー／アンダフローなし
95  b1 OVFAF12 = 0 ECNTAF12 にオーバーフローなし
96  b0 ICFF12 = 0 サブブロック F12 にインプットキャプチャの検出なし */
97  ATUF.SUBBLOCK[12].TSRF.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
98
99  /* Configure TIERF12
100 b7:4 Reserved
101 b3 OVECF12 = 0 OVFCF12 による割り込みを禁止
102 b2 OVEBF12 = 0 OVFBF12 による割り込みを禁止
103 b1 OVEAF12 = 0 OVFAF12 による割り込みを禁止
104 b0 ICEF12 = 1 ICFF12 による割り込みを許可 */
105 ATUF.SUBBLOCK[12].TIERF.BYTE = 0x01; /* ICEF12 セットによる割り込みを許可 */
106
107 /* Configure TSTRF
108 b12 STRF12 = 1 ECNTAF12、ECNTBF12、および ECNTCF12 のカウント動作を許可 */
109 ATUF.TSTRF.BIT.STRF12 = 1; /* サブブロック 12 のカウントをスタート */
110
111 /* Configure PSCR0
112 b15:10 Reserved
113 b9:0 PSC0 = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
114 ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
115
116 /* Configure IPR16
117 b15:12 ATU-F (ICIF0～ICIF3) = H'0 対応する割り込みの優先順位を 0 に設定
118 b11:8 ATU-F (ICIF4～ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
119 b7:4 ATU-F (ICIF8～ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
120 b3:0 ATU-F (ICIF12～ICIF15) = H'E 対応する割り込みの優先順位を 14 に設定 */
121 INTC.IPR16.WORD = 0x000E; /* ICIF12 の割り込み優先順位を 14 に設定 */
122
123 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
124
125 /* Configure ATUENR
126 b15:10 Reserved
127 b9 TJE = 0 タイマ J のカウント動作を停止
128 b8 THE = 0 タイマ H のカウント動作を停止
129 b7 TGE = 0 タイマ G のカウント動作を停止
130 b6 TFE = 1 タイマ F のカウント動作を許可
131 b5 TEE = 1 タイマ E のカウント動作を許可
132 b4 TDE = 0 タイマ D のカウント動作を停止
133 b3 TCE = 0 タイマ C のカウント動作を停止
134 b2 TBE = 0 タイマ B のカウント動作を停止
135 b1 TAE = 0 タイマ A のカウント動作を停止
136 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
137 ATUCTRL.ATUENR.WORD = 0x0061; /* タイマ E, F およびプリスケーラのカウントをスタート */
138
139 while(1);
140 } /* End of function main() */
141
142 /*****

```

```

143  * Function Name : ICIF12
144  * Description   : インプットキャプチャ割り込み
145  * Arguments     : none
146  * Return Value  : none
147  *****/
148  void ICIF12(void)
149  {
150  /* Configure GRAF12
151  b31:8 GRAF12[23:0] ECNTAF12 のデータ
152  b7:0 Reserved */
153  Offduty_cnt = (ATUF.SUBBLOCK[12].GRAF >> 8); /* 変数 Offduty_cnt に直前の PWM 波形の
    オフデューティの値を保存 */
154
155  /* Configure GRCF12
156  b31:8 GRCF12[23:0] ECNTCF12 のデータ
157  b7:0 Reserved */
158  PWM_cycle   = (ATUF.SUBBLOCK[12].GRCF >> 8); /* 変数 PWM_cycle に直前の PWM 波形のサ
    イクルの値を保存 */
159
160  /* Configure CDRF12
161  b31:8 CDRF0[23:0] ECNTBF12 のデータ
162  b7:0 Reserved */
163  Edge_cnt    = (ATUF.SUBBLOCK[12].CDRF >> 16); /* 変数 Edge_cnt に動作開始からのエッジ数
    を保存 */
164
165  /* Configure GRDF12
166  b31:8 GRDF12[23:0] ECNTAF12 の累積データ
167  b7:0 Reserved */
168  Timer_cnt   = (ATUF.SUBBLOCK[12].GRDF >> 8); /* 変数 Timer_cnt に動作開始からのカウ
    ント数を保存 */
169
170  /* Configure TSRF0
171  b7:4 Reserved
172  b3 OVFCF12 = 0 オーバフローフラグをクリア
173  b2 OVBF12 = 0 オーバフロー/アンダフローフラグをクリア
174  b1 OVFAF12 = 0 オーバフローフラグをクリア
175  b0 ICFF12 = 0 インプットキャプチャの検出フラグをクリア */
176  ATUF.SUBBLOCK[12].TSRF.BYTE &= 0x00; /* ICEF0 を 0 にクリア */
177  } /* End of function ICIF12() */

```

2.17.6 SH7253 グループでご使用の場合の変更点

本文 2.17 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください。

2.18 動作例 17 アップダウンイベントカウント[タイマ F]

2.18.1 概要

- 1) 二つの入力端子を用い、一方への入力エッジでカウントを、もう一方への入力レベルでそのカウントがアップカウントか、ダウンカウントかを切り替え、カウントを行います。
- 2) 一定期間毎にその時点でのカウントの値を保存し、カウントの値をクリアします。

2.18.2 使用機能説明

表 2.17に使用端子及び関連レジスタの機能割り付けを示します。

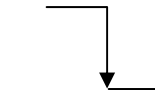
表 2.17 ATU-III機能割り付け

使用端子		機能
端子	TIF0A	外部から信号（カウント用エッジ）を入力します。
	TIF0B	外部から信号（カウント方向用レベル）を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TCRF0	タイマ F 各サブブロックの動作を設定します。
	ECNTAF0	指定したクロックバスで時間を計測します。
	ECNTBF0	二端子への入力に従ってアップ/ダウンカウント動作を行います。
	GRAF0	ECNTAF0 に対する、コンペアマッチの値を設定します。
	GRBF0	ECNTAF0 のコンペアマッチ発生時に、ECNTBF0 の値をキャプチャします。
	TSRF0	タイマ F サブブロック 0 の状態が格納されます。
PFC レジスタ	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
	PHCR	ポート H の端子機能を設定します。
INTC レジスタ	PDCR2	ポート D の端子機能を設定します。
	IPR16	コンペアマッチ割り込み ICIF0 の優先度を設定します

2.18.3 動作説明

表 2.18、図 2.33に二端子の入力に対応した動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってアップ/ダウンカウントを行います。

表 2.18 アップ/ダウンイベントカウントモード時のカウント方向

入力	カウント方向	
	アップカウント	ダウンカウント
TIFnA (入力エッジ)		
TIFnB (入力レベル)	ローレベル	ハイレベル

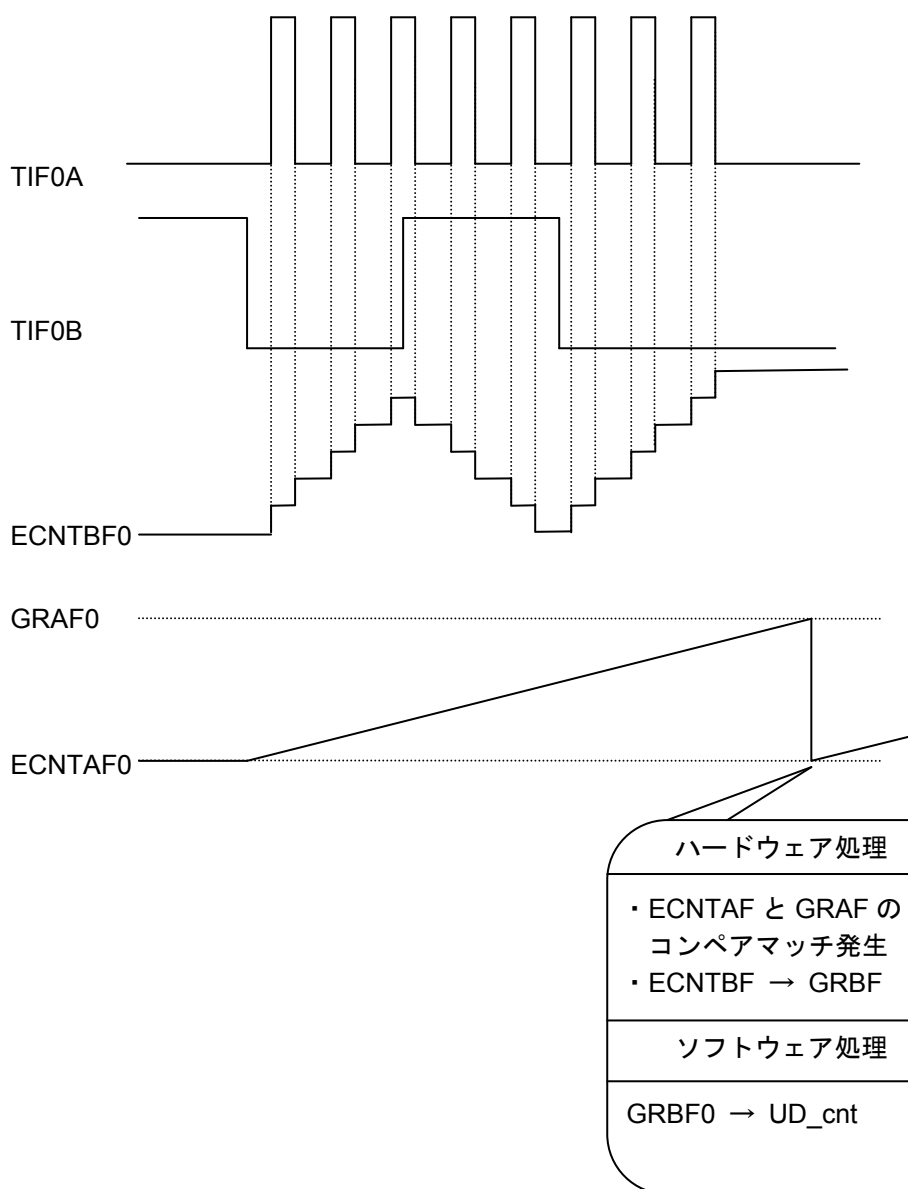


図 2.33 アップダウンイベントカウント動作原理

2.18.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
コンペアマッチ 割り込みルーチン	ICIF0	ECNTAF0、GRAF0のコンペアマッチ時に割り込み、ECNTBF0の値をGRBF0にキャプチャします。

● 使用変数の説明

変数名	機能
UD_cnt	コンペアマッチ割り込み発生時のECNTBF0の値を保存します。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PHCR	PH0端子をTIF0Aに設定します。	0x0003	メインルーチン
PDCR2	PD9端子をTIF0Bに設定します。	0x0008	
PSCR0	プリスケアラ0の分周比を50に設定します。	0x0031	
ATUENR	ATU-IIIのタイマFおよびプリスケアラのカウント動作を許可します。	0x0043	
TSTRF	タイマFサブブロック0のカウントをスタートさせます。	0x00000001	
TSRF0	サブブロック0のフラグ管理を行います。	0x00	
TCRF0	サブブロック0を「クロックバス0でカウント」「アップ/ダウンイベントカウント」「計測エッジは両エッジ」に設定します。	0x1F	
TIERF0	ECNTAF0、GRAF0のコンペアマッチ発生時の割り込みを許可します。	0x01	
GRAF0	検出するエッジの回数をFF(256)回に設定します。	0x0000FF00	
IPR16	ICIF0の割り込み優先順位を0xEに設定します。	0xE000	コンペアマッチ割 り込みルーチン
TSRF0	タイマFサブブロック0のステータスレジスタをクリアします。	0x00	

2.18.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Up/Down event count.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned short UD_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PHCR
63  b15:12 Reserved
64  b11:10 PH5MD[1:0] = 00 PH5 入出力 (ポート)
65  b9:8 PH4MD[1:0] = 00 PH4 入出力 (ポート)
66  b7:6 PH3MD[1:0] = 00 PH3 入出力 (ポート)
67  b5:4 PH2MD[1:0] = 00 PH2 入出力 (ポート)
68  b3:2 PH1MD[1:0] = 00 PH1 入出力 (ポート)
69  b1:0 PH0MD[1:0] = 11 TIF0A 入力 (ATU-III) */
70  PORTH.CR.WORD = 0x0003; /* PH0 を TIF0A(エッジ入力用端子)に設定 */
71
72  /* Configure PDCR2
73  b15:12 Reserved
74  b11:10 PD13MD[1:0] = 00 PD13 入出力 (ポート)
75  b9:8 PD12MD[1:0] = 00 PD12 入出力 (ポート)
76  b7:6 PD11MD[1:0] = 00 PD11 入出力 (ポート)
77  b5:4 PD10MD[1:0] = 00 PD10 入出力 (ポート)
78  b3:2 PD9MD[1:0] = 10 TIF0B 入力 (ATU-III)
79  b1:0 PD8MD[1:0] = 00 PD8 入出力 (ポート) */
80  PORTD.CR2.WORD = 0x0008; /* PD9 を TIF0B(エッジ入力用端子)に設定 */
81
82  /* ATU-III (4 通倍イベントカウンタ)の設定 */
83  /* Configure TCRF0
84  b7:5 CKSELF0[2:0] = 000 クロックバス 0
85  b4:2 MDF0[2:0] = 111 4 通倍イベントカウンタ
86  b1:0 EGSELEn[1:0] = 11 両エッジ */
87  ATUF.SUBBLOCK[0].TCRF.BYTE= 0x1F; /* サブブロック 0 の時間計測カウンタのクロックソース
   を「クロックバス 0」,
88  動作モードを「4 通倍イベントカウンタ」,
89  計測エッジを「両エッジ」に それぞれ設定 */
90
91  /* Configure GRAF0
92  b31:8 GRAF0[23:0] アウトプットコンペア値
93  b7:0 Reserved */

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94  ATUF.SUBBLOCK[0].GRAF = 0x0000FF00; /* 計測する期間をクロックバス 0 0xFF 回(254 周期分)
    に設定 */
95
96  /* Configure TIERF0
97  b7:4 Reserved
98  b3 OVECF0 = 0 OVFCF0 による割り込みを禁止
99  b2 OVEBF0 = 0 OVFBF0 による割り込みを禁止
100 b1 OVEAF0 = 0 OVFAF0 による割り込みを禁止
101 b0 ICEF0 = 1 ICFF0 による割り込みを許可 */
102 ATUF.SUBBLOCK[0].TIERF.BYTE = 0x01; /* ECNTAF0 のコンペアマッチ割り込み(ICIF0)を許可
    */
103
104 /* Configure TSTRF
105 b0 STRF0 = 1 ECNTAF0、ECNTBF0、および ECNTCF0 のカウント動作を許可 */
106 ATUF.TSTRF.BIT.STRF0 = 1; /* サブブロック 0 のカウントをスタート */
107
108 /* Configure PSCR0
109 b15:10 Reserved
110 b9:0 PSC0 = H'31 プリスケーラの分周比を 50(設定値は-1)に設定 */
111 ATUCTRL.PSCR0.WORD = 0x00031; /* プリスケーラの分周比を 50 に設定 */
112
113 /* Configure IPR16
114 b15:12 ATU-F (ICIF0~ICIF3) = H'E 対応する割り込みの優先順位を 14 に設定
115 b11:8 ATU-F (ICIF4~ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
116 b7:4 ATU-F (ICIF8~ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
117 b3:0 ATU-F (ICIF12~ICIF15) = H'0 対応する割り込みの優先順位を 0 に設定 */
118 INTC.IPR16.WORD = 0xE000; /* ICIF0 の割り込み優先順位を 14 に設定 */
119
120 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
121
122 /* Configure ATUENR
123 b15:10 Reserved
124 b9 TJE = 0 タイマ J のカウント動作を停止
125 b8 THE = 0 タイマ H のカウント動作を停止
126 b7 TGE = 0 タイマ G のカウント動作を停止
127 b6 TFE = 1 タイマ F のカウント動作を許可
128 b5 TEE = 0 タイマ E のカウント動作を停止
129 b4 TDE = 0 タイマ D のカウント動作を停止
130 b3 TCE = 0 タイマ C のカウント動作を停止
131 b2 TBE = 0 タイマ B のカウント動作を停止
132 b1 TAE = 1 タイマ A のカウント動作を許可
133 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
134 ATUCTRL.ATUENR.WORD = 0x0043; /* タイマ A, F およびプリスケーラのカウントをスタート */
135
136 while(1);
137 } /* End of function main() */
138
139 /*****
140 * Function Name : ICIF0
141 * Description   : コンペアマッチ割り込み

```

```
142  * Arguments      : none
143  * Return Value   : none
144  *****/
145  void ICIF0(void)
146  {
147  /* Configure TSRF0
148  b7:4 Reserved
149  b3 OVFCF0 = 0 オーバフローフラグをクリア
150  b2 OVFBF0 = 0 オーバフロー／アンダフローフラグをクリア
151  b1 OVFAF0 = 0 オーバフローフラグをクリア
152  b0 ICFF0 = 0 インプットキャプチャの検出フラグをクリア */
153  ATUF.SUBBLOCK[0].TSRF.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
154
155  /* Configure GRBF0
156  b15:0 CDRFn[23:0] ECNTBF0 のデータ */
157  UD_cnt = ATUF.SUBBLOCK[0].GRBF; /* キャプチャした ECNTBF の値を変数に保存 */
158  } /* End of function ICIF0() */
```

2.18.6 SH7253 グループでご使用の場合の変更点

本文 2.18 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください
- (2)TIF0A 端子が、SH7253 グループでは、PJ4 端子 (SH7254R グループでは、PH0 端子) に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。
- (3)TIF0B 端子が、SH7253 グループでは、PF0 端子 (SH7254R グループでは、PD9 端子) に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.18.5 サンプルプログラムリストの 70 行目を

PORTJ.CR2A.WORD=0x0001 に変更

SH7254R グループ : PHCR を 0x0003 に設定 (変更前)

SH7253 グループ : PJCR2A を 0x0001 に設定 (変更後)

2.18.5 サンプルプログラムリストの 80 行目を

PORTF.CR1A.WORD=0x0001 に変更

SH7254R グループ : PDCR を 0x0008 に設定 (変更前)

SH7253 グループ : PFCR1A を 0x0001 に設定 (変更後)

2.19 動作例 18 4 通倍イベントカウント[タイマ F]

2.19.1 概要

二つの入力端子を用い、両方の入力エッジ及び入力レベルによってアップカウントか、ダウンカウントかを切り替え、カウントを行います。

2.19.2 使用機能説明

表 2.19に使用端子及び関連レジスタの機能割り付けを示します。

表 2.19 ATU-III機能割り付け

使用端子		機能
端子	TIF0A	外部から信号を入力します。
	TIF0B	外部から信号を入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ F レジスタ	TCRF0	タイマ F 各サブブロックの動作を設定します。
	TIERF0	タイマ F 割り込み許可の設定を行います。
	ECNTAF0	指定したクロックバスで時間を計測します。
	ECNTBF0	二端子への入力に従って 4 通倍イベントカウント動作を行います。
	GRAF0	ECNTAF0 に対する、コンペアマッチの値を設定します。
	GRBF0	ECNTAF0 のコンペアマッチ発生時に、ECNTBF0 の値をキャプチャします。
	TSTRF	タイマ F 各サブブロックのカウンタの動作を設定します。
PFC レジスタ	PHCR	ポート H の端子機能を設定します。
	PDCR2	ポート D の端子機能を設定します。
INTC レジスタ	IPR16	コンペアマッチ割り込み ICIF0 の優先度を設定します

2.19.3 動作説明

表 2.20に二端子の入力に対応した動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって4 通倍イベントカウントを行います。

表 2.20 アップ/ダウンイベントカウントモード時のカウント方向

入力	カウント方向							
	アップカウント				ダウンカウント			
TIFnA (入力エッジ)	ハイレベル		ローレベル		ハイレベル		ローレベル	
TIFnB (入力レベル)		ハイレベル		ローレベル		ローレベル		ハイレベル

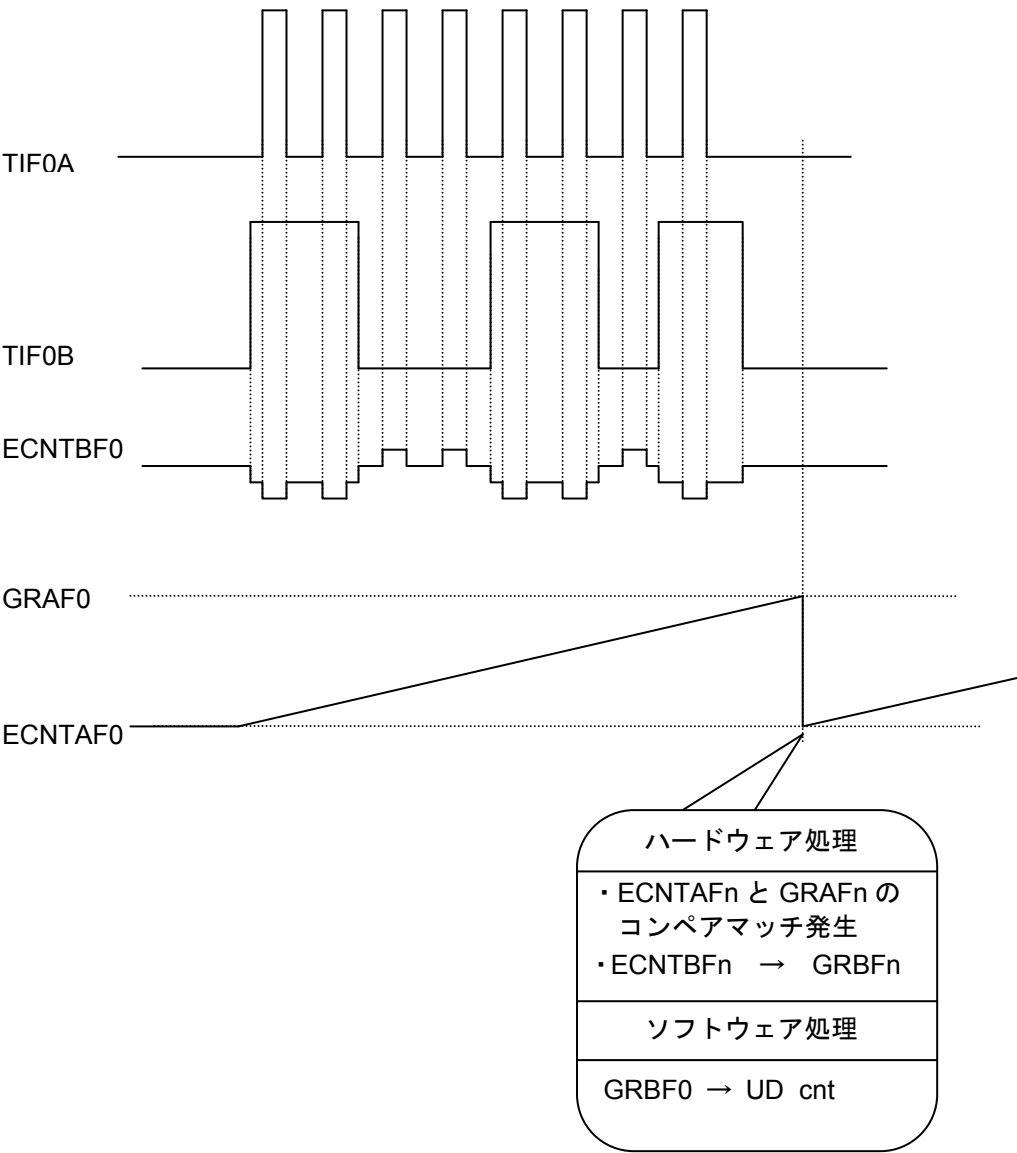


図 2.34 アップダウンイベントカウント動作原理

2.19.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
コンペアマッチ 割り込みルーチン	ICIF0	ECNTAF0、GRAF0 のコンペアマッチ時に割り込みを発生し、ECNTBF0 の値を GRBF0 にキャプチャします。

● 使用変数の説明

変数名	機能
UD_cnt	コンペアマッチ割り込み発生時の ECNTBF0 の値を保存します。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PHCR	PH0 端子を TIF0A に設定します。	0x0003	メインルーチン
PDCR2	PD9 端子を TIF0B に設定します。	0x0008	
PSCR0	プリスケアラ 0 の分周比を 50 に設定します。	0x031	
ATUENR	ATU-IIIのタイマ F およびプリスケアラのカウント動作を許可します。	0x0043	
TSTRF	タイマ F サブブロック 0 のカウントをスタートさせます。	0x00000001	
TSRF0	サブブロック 0 のフラグ管理を行います。	0x00	
TCRF0	サブブロック 0 を「クロックバス 0 でカウント」「アップ/ダウンイベントカウント」「計測エッジは両エッジ」に設定します。	0x1B	
TIERF0	ECNTAF0、GRAF のコンペアマッチ発生時の割り込みを許可します。	0x01	
GRAF0	検出するエッジの回数を FF(256)回に設定します。	0x0000FF00	
ECNTAF0	TCRF で設定したクロックバスでアップカウントを行います。	—	
ECNTBF0	二端子への入力に対応したアップ/ダウンカウントを行います。	—	
GRBF0	コンペアマッチ発生時に ECNTBF0 の値を保存します。	—	

2.19.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Event count multiplied by 4.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long UD_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58  void main(void)
59  {
60  /* ポートの設定 */
61  /* Configure PHCR
62  b15:12 Reserved
63  b11:10 PH5MD[1:0] = 00 PH5 入出力 (ポート)
64  b9:8 PH4MD[1:0] = 00 PH4 入出力 (ポート)
65  b7:6 PH3MD[1:0] = 00 PH3 入出力 (ポート)
66  b5:4 PH2MD[1:0] = 00 PH2 入出力 (ポート)
67  b3:2 PH1MD[1:0] = 00 PH1 入出力 (ポート)
68  b1:0 PH0MD[1:0] = 11 TIF0A 入力 (ATU-III) */
69  PORTH.CR.WORD = 0x0003; /* PH0 を TIF0A(エッジ入力用端子)に設定 */
70
71  /* Configure PDCR2
72  b15:12 Reserved
73  b11:10 PD13MD[1:0] = 00 PD13 入出力 (ポート)
74  b9:8 PD12MD[1:0] = 00 PD12 入出力 (ポート)
75  b7:6 PD11MD[1:0] = 00 PD11 入出力 (ポート)
76  b5:4 PD10MD[1:0] = 00 PD10 入出力 (ポート)
77  b3:2 PD9MD[1:0] = 10 TIF0B 入力 (ATU-III)
78  b1:0 PD8MD[1:0] = 00 PD8 入出力 (ポート) */
79  PORTD.CR2.WORD = 0x0008; /* PD9 を TIF0B(エッジ入力用端子)に設定 */
80
81  /* ATU-III(4 通倍イベントカウント)の設定 */
82  /* Configure TCRF0
83  b7:5 CKSELF0[2:0] = 000 クロックバス 0
84  b4:2 MDF0[2:0] = 111 4 通倍イベントカウント
85  b1:0 EGSELFn[1:0] = 11 両エッジ */
86  ATUF.SUBBLOCK[0].TCRF.BYTE= 0x1F; /* サブブロック 0 の時間計測カウンタのクロックソースを「クロッ
      クバス 0」,
87  動作モードを「4 通倍イベントカウント」,
88  計測エッジを「両エッジ」に それぞれ設定 */
89
90  /* Configure GRAF0
91  b31:8 GRAF0[23:0] アウトプットコンペア値
92  b7:0 Reserved */

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

93  ATUF.SUBBLOCK[0].GRAF = 0x0000FF00; /* 計測する期間をクロックバス 0 0xFF 回(254 周期分)
    に設定 */
94
95  /* Configure TIERF0
96  b7:4 Reserved
97  b3 OVECF0 = 0 OVFCF0 による割り込みを禁止
98  b2 OVEBF0 = 0 OVFBF0 による割り込みを禁止
99  b1 OVEAF0 = 0 OVFAF0 による割り込みを禁止
100 b0 ICEF0 = 1 ICFF0 による割り込みを許可 */
101 ATUF.SUBBLOCK[0].TIERF.BYTE = 0x01; /* ECNTAF0 のコンペアマッチ割り込み(ICIF0)を許可
    */
102
103 /* Configure TSTRF
104 b0 STRF0 = 1 ECNTAF0、ECNTBF0、および ECNTCF0 のカウント動作を許可 */
105 ATUF.TSTRF.BIT.STRF0 = 1; /* サブブロック 0 のカウントをスタート */
106
107 /* Configure PSCR0
108 b15:10 Reserved
109 b9:0 PSC0 = H'31 プリスケーラの分周比を 50(設定値は-1)に設定 */
110 ATUCTRL.PSCR0.WORD = 0x00031; /* プリスケーラの分周比を 50 に設定 */
111
112 /* Configure IPR16
113 b15:12 ATU-F (ICIF0~ICIF3) = H'E 対応する割り込みの優先順位を 14 に設定
114 b11:8 ATU-F (ICIF4~ICIF7) = H'0 対応する割り込みの優先順位を 0 に設定
115 b7:4 ATU-F (ICIF8~ICIF11) = H'0 対応する割り込みの優先順位を 0 に設定
116 b3:0 ATU-F (ICIF12~ICIF15) = H'0 対応する割り込みの優先順位を 0 に設定 */
117 INTC.IPR16.WORD = 0xE000; /* ICIF0 の割り込み優先順位を 14 に設定 */
118
119 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
120
121 /* Configure ATUENR
122 b15:10 Reserved
123 b9 TJE = 0 タイマ J のカウント動作を停止
124 b8 THE = 0 タイマ H のカウント動作を停止
125 b7 TGE = 0 タイマ G のカウント動作を停止
126 b6 TFE = 1 タイマ F のカウント動作を許可
127 b5 TEE = 0 タイマ E のカウント動作を停止
128 b4 TDE = 0 タイマ D のカウント動作を停止
129 b3 TCE = 0 タイマ C のカウント動作を停止
130 b2 TBE = 0 タイマ B のカウント動作を停止
131 b1 TAE = 1 タイマ A のカウント動作を許可
132 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
133 ATUCTRL.ATUENR.WORD = 0x0043; /* タイマ A, F およびプリスケーラのカウントをスタート */
134
135 while(1);
136 } /* End of function main() */
137
138 /*****
139 * Function Name : ICIF0
140 * Description   : インพุットキャプチャ割り込み

```

```
141  * Arguments      : none
142  * Return Value   : none
143  *****/
144  void ICIF0(void)
145  {
146  /* Configure TSRF0
147  b7:4 Reserved
148  b3 OVFCF0 = 0 オーバフローフラグをクリア
149  b2 OVFBF0 = 0 オーバフロー／アンダフローフラグをクリア
150  b1 OVFAF0 = 0 オーバフローフラグをクリア
151  b0 ICFF0 = 0 インプットキャプチャの検出フラグをクリア */
152  ATUF.SUBBLOCK[0].TSRF.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
153
154  /* Configure GRBF0
155  b15:0 CDRFn[23:0] ECNTBF0 のデータ */
156  UD_cnt = ATUF.SUBBLOCK[0].GRBF; /* キャプチャした ECNTBF の値を変数に保存 */
157  } /* End of function ICIF0() */
```

2.19.6 SH7253 グループでご使用の場合の変更点

本文 2.19 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください
- (2)TIF0A 端子が、SH7253 グループでは、PJ4 端子 (SH7254R グループでは、PH0 端子) に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。
- (3)TIF0B 端子が、SH7253 グループでは、PF0 端子 (SH7254R グループでは、PD9 端子) に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください

具体的には

2.19.5 サンプルプログラムリストの 69 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PHCR を 0x0003 に設定 (変更前)

SH7253 グループ : PJCR2A を 0x0001 に設定 (変更後)

2.19.5 サンプルプログラムリストの 79 行目を

PORTF.CR1A.WORD=0x0001 に変更

SH7254R グループ : PDCR を 0x0008 に設定 (変更前)

SH7253 グループ : PFCR1A を 0x0001 に設定 (変更後)

2.20 動作例 19 コンペアマッチ割り込み発生[タイマ G]

2.20.1 概要

タイマカウントがコンペアマッチの値と一致した時、コンペアマッチ割り込みを発生させます。

2.20.2 使用機能説明

表 2.21に関連レジスタの機能割り付けを示します。

表 2.21 ATU-III機能割り付け

関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ G レジスタ	TCRG0	タイマ G の各機能を設定します。
	OCRG0	コンペアマッチ値を設定します。
	TCNTG0	TCRG で設定したクロックバスでタイマカウントを行います。
	TSTRG	タイマ G サブクロック 0 のタイマカウンタをスタートさせます。
	TSRG0	オーバフロー、コンペアマッチが発生した場合にフラグがセットされます。
INTC レジスタ	IPR20	CMIG0 のコンペアマッチ割り込みの優先度を設定します。

2.20.3 動作説明

図 2.35に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってコンペアマッチを発生させます。タイマ G のカウント(TCNTG0)数は、コンペアマッチが発生するとクリアされます。

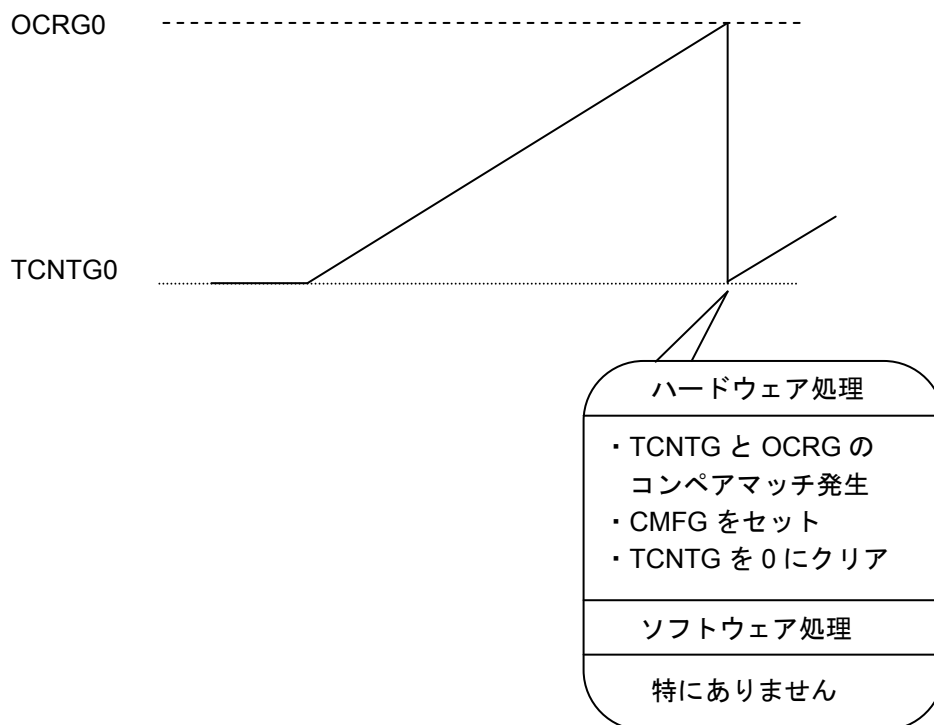


図 2.35 コンペアマッチ動作原理

2.20.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。

- 使用変数の説明

本タスクでは変数は使用していません。

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	メインルーチン
ATUENR	ATU-IIIのチャネル G 及びプリスケアラのカウント動作を許可します。	0x0081	
TSTRG	タイマ G サブブロック 0 のカウント動作を許可します。	0x3F	
TCRG0	「クロックバス 0 でカウント」「コンペアマッチパルスを出力しない」「コンペアマッチ時の割り込みを許可」に設定します。	0x01	
OCRG0	TCNTG0 とのコンペアマッチの値を設定します。	0x0400	
TSRG0	オーバフロー、コンペアマッチフラグをクリアします。	0x00	
IPR20	CMIF0 の割り込み優先順位を 15 に設定します。	0xF000	

2.20.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Compare match interrupt occurs.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments    : none
55  * Return Value  : none
56  *****/
57
58  void main(void); /* メインルーチン */
59
60  /*****
61  * Function Name : main
62  * Description   : The main loop
63  * Arguments    : none
64  * Return Value  : none
65  *****/
66  void main(void)
67  {
68  /* ATU-IIIの設定 */
69  /* Configure TCRG
70  b7 Reserved
71  b6:4 CKSELG0 = 0 クロックバス 0 をクロックソースに指定
72  b3:2 Reserved
73  b1 CMPOEG0 = 0 TCNTG0 と OCRG0 のコンペアマッチでパルス出力しない
74  b0 CMFG0 = 1 CMFG0 による割り込みを許可 */
75  ATUG.SUBBLOCK[0].TCRG.BYTE = 0x01;
76
77  /* Configure OCRG
78  b15:0 OCRG0 = H'0400 コンペアマッチの値を設定 */
79  ATUG.SUBBLOCK[0].OCRG = 0x0400; /* タイマカウンタ 0 のコンペアマッチの値を 0x0400(1024)
   に設定 */
80
81  /* Configure TSRG
82  b7:2 Reserved
83  b1 OVFG0 = 0 オーバフローフラグをクリア
84  b0 CMFG0 = 0 コンペアマッチフラグをクリア */
85  ATUG.SUBBLOCK[0].TSRG.BYTE &= 0x00; /* 各ステータスフラグをクリア */
86
87  /* Configure TSTRG
88  b7:6 Reserved
89  b5 STRG5 = 1 TCNTG5 のカウント動作を許可
90  b4 STRG4 = 1 TCNTG4 のカウント動作を許可
91  b3 STRG3 = 1 TCNTG3 のカウント動作を許可
92  b2 STRG2 = 1 TCNTG2 のカウント動作を許可
93  b1 STRG1 = 1 TCNTG1 のカウント動作を許可

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94  b0 STRG0 = 1 TCNTG0 のカウント動作を許可 */
95  ATUG.TSTRG.BYTE = 0x3F; /* タイマ G サブブロック 0 のタイマカウンタをスタート */
96
97  /* Configure PSCR0
98  b15:10 Reserved
99  b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1) に設定 */
100 ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分周比を 10 に設定 */
101
102 /* Configure IPR20
103 b15:12 ATU-G(CMIG0-CMIG3) = H'F 対応する割り込みの優先順位を 15 に設定
104 b11:8 ATU-G(CMIG4,CMIG5) = H'0 対応する割り込みの優先順位を 0 に設定
105 b7:4 ATU-H(CMIH) = H'0 対応する割り込みの優先順位を 0 に設定
106 b3:0 Reserved */
107 INTC.IPR20.WORD = 0xF000; /* CMIG0 の割り込み優先順位を 15 に設定 */
108
109 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
110
111 /* Configure ATUENR
112 b15:10 Reserved
113 b9 TJE = 0 タイマ J のカウント動作を停止
114 b8 THE = 0 タイマ H のカウント動作を停止
115 b7 TGE = 1 タイマ G のカウント動作を許可
116 b6 TFE = 0 タイマ F のカウント動作を停止
117 b5 TEE = 0 タイマ E のカウント動作を停止
118 b4 TDE = 0 タイマ D のカウント動作を停止
119 b3 TCE = 0 タイマ C のカウント動作を停止
120 b2 TBE = 0 タイマ B のカウント動作を停止
121 b1 TAE = 0 タイマ A のカウント動作を停止
122 b0 PSCE = 1 プリスケーラのカウント動作を許可 */
123 ATUCTRL.ATUENR.WORD = 0x0081; /* タイマ G およびプリスケーラのカウントをスタート */
124
125 while(1);
126 } /* End of function main() */
127
128 /*****
129 * Function Name : CMIG0
130 * Description   : コンペアマッチ割り込み
131 * Arguments     : none
132 * Return Value  : none
133 *****/
134 void CMIG0(void)
135 {
136  /* Configure TSRG
137  b7:2 Reserved
138  b1 OVFG0 = 0 オーバフローフラグをクリア
139  b0 CMFG0 = 0 コンペアマッチフラグをクリア */
140  ATUG.SUBBLOCK[0].TSRG.BYTE &= 0x00; /* 各ステータスフラグをクリア */
141  } /* End of function CMIG0() */

```

2.20.6 SH7253 グループでご使用の場合の変更点

本文 2.20 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください。

2.21 動作例 20 コンペアマッチ発生による他モジュールの起動[タイマ G]

2.21.1 概要

コンペアマッチ発生時に 1 サイクルの負論理パルス信号を生成し、その信号をトリガとして A/D 変換器を起動します。

2.21.2 使用機能説明

表 2.22に関連レジスタの機能割り付けを示します。

表 2.22 ATU-III機能割り付け

関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR1	プリスケアラ 1 の分周比を設定します。
タイマ G レジスタ	TCRG1	タイマ G の各機能を設定します。
	OCRG1	タイマ G のコンペアマッチの値を設定します。
	TCNTG1	TCRG で設定したクロックバスでタイマカウントを行います。
	TSTRG	タイマ G のタイマカウンタをスタートします。
	TSRG1	オーバーフロー、コンペアマッチが発生した場合にフラグがセットされます。
A/D 変換機 レジスタ	ADCSR0	A/D 変換機全体の設定を行います。
	ADCER0	変換したデータを保存するレジスタに関する設定を行います。
	ADANS0	A/D 変換対象チャネルを選択します。
	ADTRE0	割り込み変換要求を禁止/許可します。
	ADTRF0	割り込みトリガ終了フラグです。
	ADREF0	A/D 変換に関するフラグがセットされます。
INTC レジスタ	IPR20	CMIG1 のコンペアマッチ割り込みの優先度を設定します。

2.21.3 動作説明

図 2.36に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってパルスの周期測定を行います。コンペアマッチが発生するとコンペアマッチパルスが負論理パルスとして生成されます。この時生成された負論理パルスをトリガとして A/D 変換を行います。

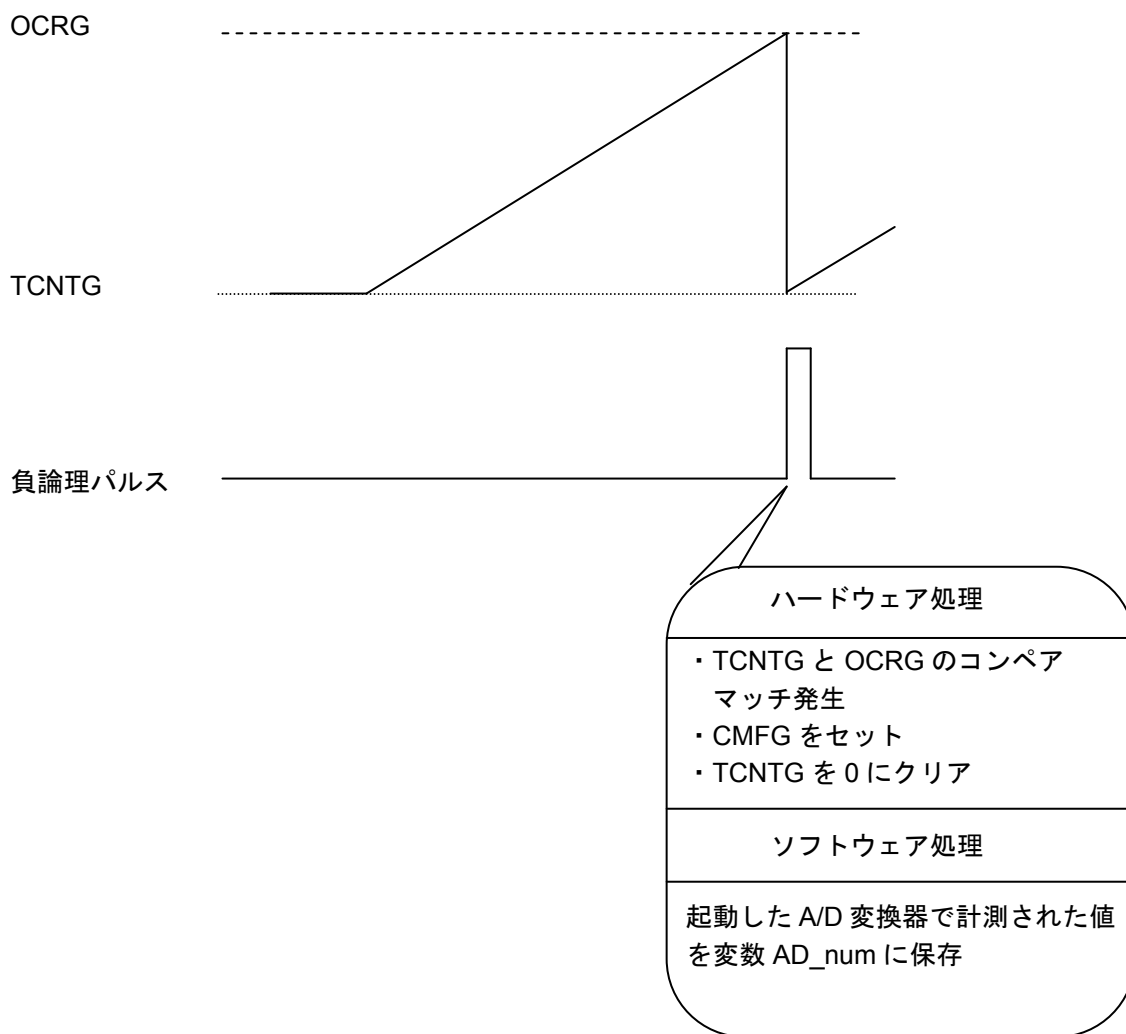


図 2.36 パルスの High 幅計測動作原理

2.21.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
A/D 設定ルーチン	AD_init	A/D 変換器の初期設定を行います。
コンペアマッチ割り込み	CMIG1	TCNTG1 と OCRG1 のコンペアマッチで割り込み、A/D 変換器で変換した値を保存します。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール名
AD_num	A/D 変換器で変換した値を保存します。	unsigned long	コンペアマッチ 割り込み

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PSCR1	プリスケアラ 1 の分周比を 10 に設定します。	0x0009	メインルーチン
ATUENR	ATU-IIIのチャンネル G 及びプリスケアラのカウント動作を許可します。	0x0081	
TSTRG	タイマ G サブブロック 0 のカウント動作を許可します。	0x02	
TCRG1	「クロックバス 0 でカウント」「コンペアマッチパルスを出力する」「コンペアマッチ時の割り込みを許可」に設定します。	0x03	
OCRG1	TCNTG1 とのコンペアマッチの値を設定します。	0x0400	
TSRG1	オーバフローフラグ、コンペアマッチフラグをクリアします。	0x00	
IPR20	コンペアマッチ割り込み(CMIG1)の優先度を設定します。	0xF000	
ADCSR0	「外部トリガによるスキャン起動を許可」「トリガは ATU-IIIのタイマトリガを使用」に設定します。	0xC2	A/D 変換器 設定ルーチン
ADCER0	「データレジスタのフォーマットは右詰」「オフセットキャンセルの処理を行う」「割り込み変換のトリガソースを ATU-IIIのタイマ G1 トリガ」に設定します。	0x8001	
ADANS0	チャンネル 0 を変換に使用するように設定します。	0x0001	
ADTRE0	チャンネル 0 への割り込み要求を許可します。	0x0001	
ADREF0	A/D 変換に関するフラグがセットされます。	0x00	
ADTRF0	A/D 割り込みが終了した際にフラグがセットされます。	0x00	

2.21.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Other modules start with compare match.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45
46 /*****

```

```

47 Private global variables and functions
48 *****/
49 unsigned long AD_num;
50 void main(void); /* メインルーチン */
51 void AD_init(void); /* A/D 変換設定ルーチン */
52
53 /*****
54 * Function Name : main
55 * Description   : The main loop
56 * Arguments     : none
57 * Return Value  : none
58
59 *****/
60 void main(void)
61 {
62 /* ATU-III の設定 */
63 /* Configure TCRG
64 b7:2 Reserved
65 b6:4 CKSELG0 = 0 クロックバス 0 をクロックソースに指定
66 b3:2 Reserved
67 b1 CMP0EG0 = 1 TCNTG0 と OCRG0 のコンペアマッチでパルスを出力
68 b0 CMFG0 = 1 CMFG0 による割り込みを許可 */
69 ATUG.SUBBLOCK[1].TCRG.BYTE = 0x03; /* クロックバス 1 を使用、コンペアマッチ時にコンペア
    マッチパルスを出力、
70     コンペアマッチ時の割り込み動作を許可 */
71 /* Configure OCRG
72 b15:0 OCRG0 = H'0400 コンペアマッチの値を設定 */
73 ATUG.SUBBLOCK[1].OCRG = 0x0400; /* タイマカウンタ 1 のコンペアマッチの値を 0x0400(1024)
    に設定 */
74
75 /* Configure TSRG
76 b7:2 Reserved
77 b1 OVFG0 = 0 オーバフローフラグをクリア
78 b0 CMFG0 = 0 コンペアマッチフラグをクリア */
79 ATUG.SUBBLOCK[1].TSRG.BYTE &= 0x00; /* 各ステータスフラグをクリア */
80
81 /* Configure TSTRG
82 b7:6 Reserved
83 b5 STRG5 = 0 TCNTG5 のカウント動作を停止
84 b4 STRG4 = 0 TCNTG4 のカウント動作を停止
85 b3 STRG3 = 0 TCNTG3 のカウント動作を停止
86 b2 STRG2 = 0 TCNTG2 のカウント動作を停止
87 b1 STRG1 = 1 TCNTG1 のカウント動作を許可
88 b0 STRG0 = 0 TCNTG0 のカウント動作を停止 */
89 ATUG.TSTRG.BYTE = 0x02; /* タイマ G サブブロック 1 のタイマカウンタをスタート */
90
91 /* Configure PSCR0
92 b15:10 Reserved
93 b9:0 PSCn = H'9 プリスケアラの分周比を 10(設定値は-1) に設定 */

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

94  ATUCTRL.PSCR1.WORD = 0x0009;    /* プリスケアラの分周比を 10 に設定 */
95
96  /* A/D 変換の設定 */
97  AD_init(); /* A/D 変換設定ルーチン */
98
99  /* Configure IPR20
100 b15:12 ATU-G(CMIG0-CMIG3) = H'F 対応する割り込みの優先順位を 15 に設定
101 b11:8 ATU-G(CMIG4,CMIG5) = H'0 対応する割り込みの優先順位を 0 に設定
102 b7:4 ATU-H(CMIH) = H'0 対応する割り込みの優先順位を 0 に設定
103 b3:0 Reserved */
104 INTC.IPR20.WORD = 0xF000;    /* CMIG1 の割り込み優先順位を 15 に設定 */
105
106 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
107
108 /* Configure ATUENR
109 b15:10 Reserved
110 b9 TJE = 0 タイマ J のカウント動作を停止
111 b8 THE = 0 タイマ H のカウント動作を停止
112 b7 TGE = 1 タイマ G のカウント動作を許可
113 b6 TFE = 0 タイマ F のカウント動作を停止
114 b5 TEE = 0 タイマ E のカウント動作を停止
115 b4 TDE = 0 タイマ D のカウント動作を停止
116 b3 TCE = 0 タイマ C のカウント動作を停止
117 b2 TBE = 0 タイマ B のカウント動作を停止
118 b1 TAE = 0 タイマ A のカウント動作を停止
119 b0 PSCE = 1 プリスケアラのカウント動作を許可 */
120 ATUCTRL.ATUENR.WORD = 0x0081; /* タイマ G およびプリスケアラのカウントをスタート */
121
122 while(1);
123 } /* End of function main() */
124
125 /*****
126 * Function Name : AD_init
127 * Description   : A/D 変換器設定
128 * Arguments    : none
129 * Return Value  : none
130 *****/
131 void AD_init(void)
132 {
133 /* Configure ADCSR
134 b7 ADST = 1 スキャン変換開始
135 b6 ADCS = 1 連続スキャンモード
136 b5 Reserved
137 b4 ADIE = 0 スキャン終了後の ADI 割り込み発生の禁止
138 b3:2 Reserved
139 b1 TRGE = 1 外部トリガまたは ATU-III のタイマトリガによるスキャン変換の起動を許可
140 b0 EXTRG = 0 ATU-III のタイマトリガによるスキャン変換の起動 */
141 ADCA.ADCSR.BYTE = 0xC2; /* 外部トリガによるスキャン変換の起動を許可
142     外部トリガは ATU-III のタイマトリガを使用 */
143 /* Configure ADCER

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

144  b15 ADRFMT = 1 A/D データレジスタのフォーマットを右詰めに設定
145  b14:12 Reserved
146  b11 DIAGM = 0 A/D 変換器の自己診断を行わない
147  b10 DIAGLD = 0 自己診断は自動的にローテーションして実施
148  b9:8 DIAGVAL = 0 リザーブ(未設定状態)
149  b7 CKS = 0 A/D 変換ステート数は 50 ステート
150  b6:1 Reserved
151  b0 ITTRGS = 1 ATU-III のタイマ G1 トリガで AN0 の割り込み変換の起動を行う */
152  ADCA.ADCER.WORD = 0x8001; /* データレジスタのフォーマットは右詰め
153      オフセットキャンセル処理の要求を行う
154      割り込み変換のトリガソースを ATU-III のタイマ G1 トリガに設定 */
155
156  /* Configure ADANS0
157  b15:0 ANSn = H'0001 スキャン変換用チャンネル ANS0 を使用 */
158  ADCA.ADANS0.WORD = 0x0001; /* チャンネル 0 を変換に使用 */
159
160  /* Configure ADTRE
161  b15:0 ANSn = H'0001 AN0 に ATU-III タイマ又はソフトトリガからの割り込み要求許可 */
162  ADCA.ADTRE.WORD = 0x0001; /* チャンネル 0 への割り込み要求を許可 */
163
164  /* Configure ADREF
165  b7 ADSCACT = 0 スキャン変換中フラグをクリア
166  b6 ADITACT = 0 割り込み変換中フラグをクリア
167  b5:1 Reserved
168  b0 ADF = 0 スキャン変換終了状態をクリア */
169  ADCA.ADREF.BYTE &= 0x00; /* 各変換ステータスフラグをクリア (アイドル中) */
170
171  /* Configure ADTRF
172  b15:0 ADTFn = 0 割り込み変換終了状態をクリア */
173  ADCA.ADTRF.WORD &= 0x0000; /* 各割り込みトリガ終了フラグをクリア */
174  } /* End of function AD_init() */
175
176  /*****
177  * Function Name : CMIG1
178  * Description   : コンペアマッチ割り込み
179  * Arguments     : none
180  * Return Value  : none
181  *****/
182  void CMIG1(void)
183  {
184  /* Configure TSRG
185  b7:2 Reserved
186  b1 OVFG0 = 0 TCNTG0 オーバフローフラグをクリア
187  b0 CMFG0 = 0 コンペアマッチフラグをクリア */
188  ATUG.SUBBLOCK[1].TSRG.BYTE &= 0x00; /* 各ステータスフラグをクリア */
189  AD_num = ADCA.ADR0; /* AD 変換器で保存した値を変数 AD_num に保存 */
190  } /* End of function CMIG1() */

```

2.21.6 SH7253 グループでご使用の場合の変更点

本文 2.21 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください。

2.22 動作例 21 コンペアマッチ発生による他モジュールの起動[タイマ G]

2.22.1 概要

- 1) コンペアマッチ発生時に、1 サイクルの負論理パルス信号を生成し、A/D 変換器を起動します。
- 2) コンペアマッチ割り込みで DMAC を起動し、A/D 変換した値を転送します。
- 3) DMAC の転送終了割り込みで DMAC の送信先アドレスを加算します。

2.22.2 使用機能説明

表 2.23に関連レジスタの機能割り付けを示します。

表 2.23 ATU-III機能割り付け

関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ G レジスタ	TCRG1	タイマ G の各機能を設定します。
	OCRG1	タイマ G のコンペアマッチ値を設定します。
	TCNTG1	TCRG で設定したクロックバスでタイマカウントを行います。
	TSRG1	オーバーフロー、コンペアマッチが発生した場合にフラグがセットされます。
A/D 変換機 レジスタ	ADCSR0	A/D 変換機全体の設定を行います。
	ADCER0	変換したデータを保存するレジスタに関する設定を行います。
	ADREF0	A/D 変換に関するフラグがセットされます。
	ADTRF0	割り込み変換が終了した際にフラグがセットされます。
	ADANS0	変換するチャンネルを設定します。
	ADTRE0	各チャンネルへの割り込み要求の許可を設定します。
DMAC レジスタ	CHCR2	DMAC の各機能設定を行います。
	DMARS2	拡張リソースを設定します。
	SAR2	転送元のアドレスを設定します。
	RSAR2	リロード時の転送元アドレスを設定します。
	DAR2	転送先のアドレスを設定します。
	RDAR2	リロード時の転送先アドレスを設定します。
	DMATCR2	転送時のデータの転送回数を設定します。
	RDMATCR2	リロード時の転送回数を設定します。
	CHFR2	DMAC の転送終了フラグが格納されます。
	DMAOR	DMAC の動作許可を設定します。
INTC レジスタ	IPR03	DMAC の転送終了割り込みの優先度を設定します。

2.22.3 動作説明

図 2.37に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってパルスの周期測定を行います。タイマ G のカウント開始前に、A/D 変換器はタイマ G コンペアマッチ時に発生する負論理パルスで起動し、DMAC はタイマ G のコンペアマッチで起動するように設定します。

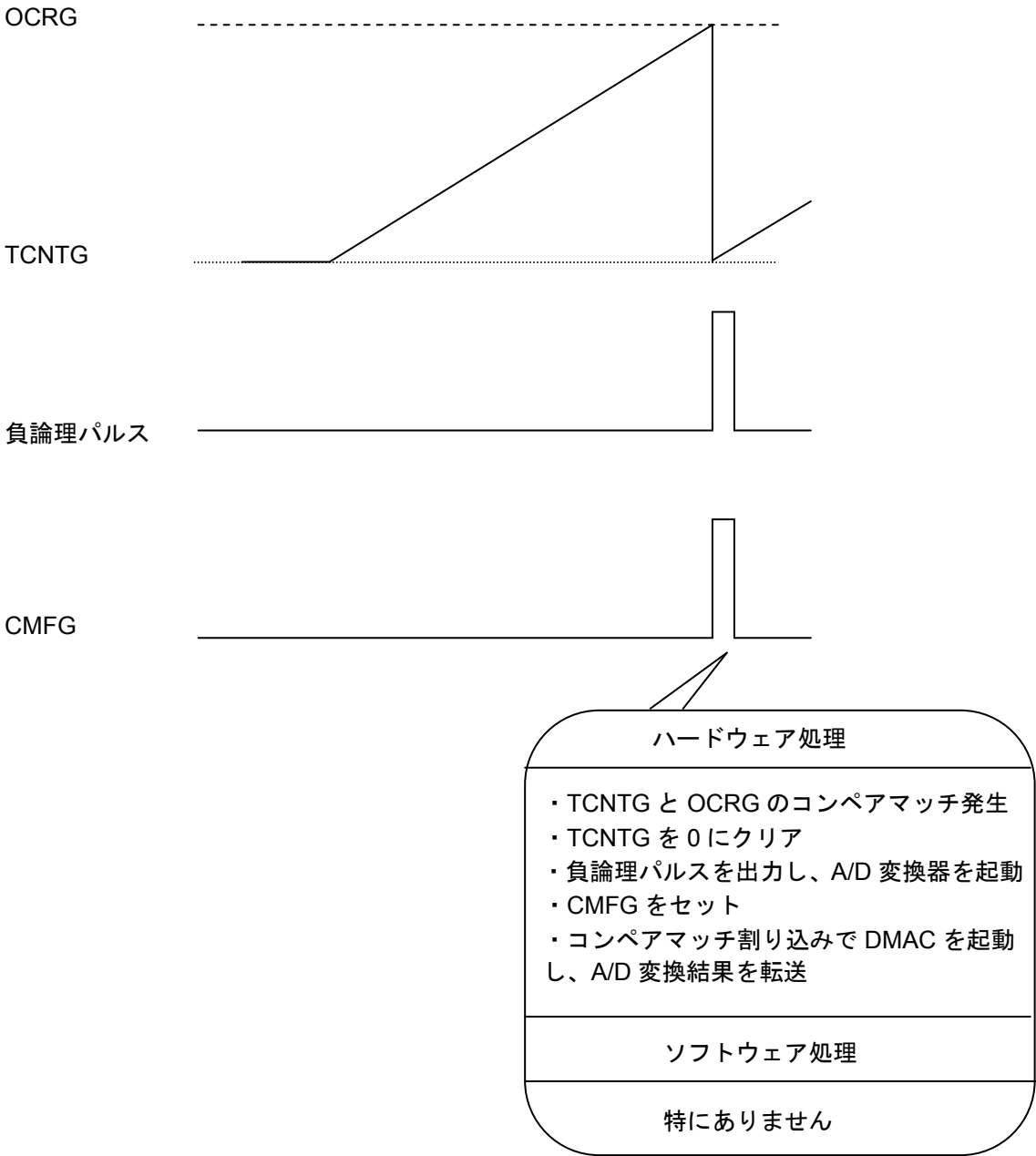


図 2.37 コンペアマッチからの A/D 変換器起動の動作原理

2.22.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
A/D 設定ルーチン	AD_init	A/D 変換器の初期設定を行います。
DMAC 設定ルーチン	DMA_init	DMA 転送の初期設定を行います。
DMA 転送終了 割り込み	DEI2	DMA 転送終了で割り込み。

● 使用変数の説明

本タスクでは変数は使用していません。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	メインルーチン
ATUENR	ATU-IIIチャンネル G 及びプリスケアラのカウンタ動作を許可します。	0x0081	
TSTRG	タイマ G サブブロック 0 のカウンタ動作を許可します。	0x01	
TSRG1	オーバフローフラグと、コンペアマッチフラグをクリアします。	0x00	
TCRG1	「クロックバス 0 でカウンタ」「コンペアマッチパルスを出力する」「コンペアマッチ時の割り込みを許可」に設定します。	0x03	
OCRG1	TCNTG0 とのコンペアマッチの値を設定します。	0x0400	
ADCSR0	「外部トリガによるスキップ起動を許可」「トリガは ATU-III のタイマトリガを使用」に設定します。	0x82	A/D 変換器設定ルーチン
ADCER0	「データレジスタのフォーマットは右詰」「オフセットキャンセルの処理を行う」「割り込み変換のトリガソースを ATU-III のタイマ G1 トリガ」に設定します。	0x8001	
ADANS0	チャンネル 0 を変換に使用するよう設定します。	0x0001	
ADTRE0	チャンネル 0 への割り込み要求を許可します。	0x0001	
ADREF0	A/D 変換に関するフラグがセットされます。	0x00	
ADTRF0	A/D 割り込みが終了した際にフラグがセットされます。	0x0000	
IPR20	コンペアマッチ割り込み(CMIG0)の優先度を設定します。	0xF000	DMAC 設定ルーチン
CHCR2	DMAC の転送モード設定としてワード転送、バースト転送、転送元/転送先のアドレスを設定します。	0x1000082C	
DMARS2	CH2 の拡張リソースを ATU-III G1 に設定します。	0x0047	
SAR2, RSAR2	転送元として ADR0 のアドレスを設定します。	0xFFFFFE840	
DAR2, RADR2	転送先として任意 RAM 領域のアドレスを設定します。	0xFFF82000	
DMATCR2, RDMATCR2	データの転送回数を 1 回に設定します。	0x0001	
IPR03	DMA 転送終了割り込み(DEI2)の優先度を設定します。	0x00F0	

SH7253 SH7254R グループ

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

CHFR2	DMA の終了フラグをクリアします。	HE = 0 TE = 0	DMA 転送終了 割り込み
TSRG1	タイマ G のコンペアマッチフラグをクリアします。	0x00	

2.22.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Other modules start with compare match.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50  void AD_init(void); /* A/D 変換設定ルーチン */
51  void DMA_init(void); /* DMAC 設定ルーチン */
52
53  /*****
54  * Function Name : main
55  * Description   : The main loop
56  * Arguments     : none
57  * Return Value  : none
58  *****/
59  void main(void)
60  {
61  /* ATUⅢの設定 */
62  /* Configure TCRG
63  b7: Reserved
64  b6:4 CKSELG0 = 0 クロックバス 0 をクロックソースに指定
65  b3:2 Reserved
66  b1 CMP0EG0 = 1 TCNTG0 と OCRG0 のコンペアマッチでパルス出力
67  b0 CMFG0 = 1 CMFG0 による割り込みを許可 */
68  ATUG.SUBBLOCK[1].TCRG.BYTE = 0x03; /* クロックバス 0 を使用、コンペアマッチ時にコンペアマッ
チパルスを出力
69  コンペアマッチ時の割り込み動作を許可 */
70  /* Configure OCRG
71  b15:0 OCRG0 = H'0400 コンペアマッチの値を設定 */
72  ATUG.SUBBLOCK[1].OCRG = 0x0400; /* タイマカウンタ 0 のコンペアマッチの値を
0x0400(1024) に設定 */
73
74  /* Configure TSRG
75  b7:2 Reserved
76  b1 OVFG0 = 0 TCNTG0 オーバフローフラグをクリア
77  b0 CMFG0 = 0 コンペアマッチフラグをクリア */
78  ATUG.SUBBLOCK[1].TSRG.BYTE &= 0x00; /* 各ステータスフラグをクリア */
79
80  /* Configure TSTRG
81  b7:6 Reserved
82  b5 STRG5 = 1 TCNTG5 のカウント動作を停止
83  b4 STRG4 = 1 TCNTG4 のカウント動作を停止
84  b3 STRG3 = 1 TCNTG3 のカウント動作を停止
85  b2 STRG2 = 1 TCNTG2 のカウント動作を停止
86  b1 STRG1 = 1 TCNTG1 のカウント動作を許可
87  b0 STRG0 = 1 TCNTG0 のカウント動作を停止 */
88  ATUG.TSTRG.BYTE = 0x02; /* タイマ G サブブロック 1 のタイマカウンタをスタート */
89  /* Configure PSCR0
90  b15:10 Reserved
91  b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1) に設定 */
92  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラの分週比を 10 に設定 */

```



```

93
94  /* 他モジュール(AD,DMAC)の設定 */
95  AD_init(); /* AD 変換器設定ルーチンへ */
96  DMA_init(); /* DMAC 設定ルーチンへ */
97
98  set_imask(0); /* 割り込みマスケレベルを 0 に設定 */
99
100 /* Configure ATUENR
101 b15:10 Reserved
102 b9 TJE = 0 タイマ J のカウント動作を停止
103 b8 THE = 0 タイマ H のカウント動作を停止
104 b7 TGE = 1 タイマ G のカウント動作を許可
105 b6 TFE = 0 タイマ F のカウント動作を停止
106 b5 TEE = 0 タイマ E のカウント動作を停止
107 b4 TDE = 0 タイマ D のカウント動作を停止
108 b3 TCE = 0 タイマ C のカウント動作を停止
109 b2 TBE = 0 タイマ B のカウント動作を停止
110 b1 TAE = 0 タイマ A のカウント動作を停止
111 b0 PSCE = 1 プリスケアラのカウント動作を許可 */
112 ATUCTRL.ATUENR.WORD = 0x0081; /* タイマ G およびプリスケアラのカウントをスタート */
113
114 while(1);
115 } /* End of function main() */
116
117 /*****
118 * Function Name : AD_init
119 * Description    : A/D 変換器設定ルーチン
120 * Arguments      : none
121 * Return Value   : none
122 *****/
123 void AD_init(void)
124 {
125  /* Configure ADCSR
126  b7 ADST = 1 スキャン変換開始
127  b6 ADCS = 1 1 サイクルスキャンモード
128  b5 Reserved
129  b4 ADIE = 0 スキャン終了後の ADI 割り込み発生の禁止
130  b3:2 Reserved
131  b1 TRGE = 1 外部トリガまたは ATU-III のタイマトリガによるスキャン変換の起動を許可
132  b0 EXTRG = 0 ATU-III のタイマトリガによるスキャン変換の起動 */
133  ADCA.ADCSR.BYTE = 0x82; /* 外部トリガによるスキャン変換の起動を許可
134      外部トリガは ATU-III のタイマトリガを使用 */
135  /* Configure ADCER
136  b15 ADRFMT = 1 A/D データレジスタのフォーマットを右詰めに設定
137  b14:12 Reserved
138  b11 DIAGM = 0 A/D 変換器の自己診断を行わない
139  b10 DIAGLD = 0 自己診断は自動的にローテーションして実施
140  b9:8 DIAGVAL = 0 リザーブ(未設定状態)
141  b7 CKS = 0 A/D 変換ステート数は 50 ステート
142  b6:1 Reserved

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

143  b0 ITTRGS = 1 ATU-III のタイマ G1 トリガで AN0 の割り込み変換の起動を行う */
144  ADCA.ADCER.WORD = 0x8001; /* データレジスタのフォーマットは右詰め
145      オフセットキャンセル処理の要求を行う
146      割り込み変換のトリガソースを ATU-III のタイマ G1 トリガに設定 */
147
148  /* Configure ADANS0
149  b15:0 ANSn = H'0001 スキャン変換用チャンネル ANS0 を使用 */
150  ADCA.ADANS0.WORD = 0x0001; /* チャンネル 0 を変換に使用 */
151
152  /* Configure ADTRE
153  b15:0 ANSn = H'0001 AN0 に ATU-III タイマ又はソフトトリガからの割り込み要求許可 */
154  ADCA.ADTRE.WORD = 0x0001; /* チャンネル 0 への割り込み要求を許可 */
155
156  /* Configure ADREF
157  b7 ADSCACT = 0 スキャン変換中フラグをクリア
158  b6 ADITACT = 0 割り込み変換中フラグをクリア
159  b5:1 Reserved
160  b0 ADF = 0 スキャン変換終了状態をクリア */
161  ADCA.ADREF.BYTE &= 0x00; /* 各変換ステータスフラグをクリア */
162
163  /* Configure ADTRF
164  b15:0 ADTFn = 0 割り込み変換終了状態をクリア */
165  ADCA.ADTRF.WORD &= 0x0000; /* 各割り込みトリガ終了フラグをクリア */
166  } /* End of function AD_init() */
167
168  /*****
169  * Function Name : DMA_init
170  * Description   : DMAC 設定ルーチン
171  * Arguments    : none
172  * Return Value  : none
173  *****/
174  void DMA_init(void)
175  {
176      /* Configure CHCR
177      b31:30 TC = 10 一回の転送要求で、DMATCR の設定回数分 n 転送
178      b29 Reserved
179      b28 RLD1 = 0 リロード機能 1 無効
180      b27:26 RLD2 = 00 リロード機能 2 無効
181      b25 Reserved
182      b24 IFT = 0 Integer→Floating-point 変換機能は無効
183      b23:19 Reserved
184      b18 HIE = 0 DMATCR =(転送前にセットした DMATCR)/2 の時、CPU への割り込み要求を禁止
185      b17:16 Reserved
186      b15:14 DM = 00 デスティネーションアドレスは固定 (16 バイト単位転送時は設定禁止)
187      b13:12 SM = 00 ソースアドレスは固定 (16 バイト単位転送時は設定禁止)
188      b11:8 RS = 1000 DMA 拡張リソースセクタ
189      b7:6 Reserved
190      b5 TB = 1 バーストモードで転送
191      b4:3 TS = 01 ワード単位で転送
192      b2 IE = 1 割り込み要求を許可

```

```

193  b1 Reserved
194  b0 DE = 0 DMA 転送を禁止 */
195  DMAC2.CHCR.LONG = 0x1000082C; /* 転送サイズ:ワード リソース:拡張リソースセクタ
196      DMAC 転送終了割り込みを許可
197      バースト転送 転送先のアドレスを固定 */
198  /* Configure DMARS
199  b15:10 CH3 MID = H'0 (default)DMA 転送要求元を ATU-III_C0 に指定
200  b9:8 CH3 RID = H'0 (default)DMA 転送要求元を ATU-III_C0 に指定
201  b7:2 CH2 MID = 0100 DMA 転送要求元を ATU-III_G1 に指定
202  b1:0 CH2 RID = 0111 DMA 転送要求元を ATU-III_G1 に指定 */
203  DMACCOM.DMARSx[1].DMARS.WORD = 0x0047; /* CH2 の拡張リソースを ATU-III_G1 に設定 */
204
205  /* Configure SAR0
206  b31:0 DMA ソースアドレス */
207  DMAC2.SAR = 0xFFFFE840; /* 転送元のアドレスを指定(AD0 の値) */
208
209  /* Configure DAR
210  b31:0 DMA デスティネーションアドレス */
211  DMAC2.DAR = 0xFFFF82000; /* 転送先のアドレスを指定 */
212
213  /* Configure DMATCR
214  b31:24 Reserved
215  b23:0 DMA 転送回数 = 1 */
216  DMAC2.DMATCR = 0x0001; /* DMA 転送の回数を設定 */
217
218  /* Configure RSAR
219  b31:0 DMA リロード時のソースアドレス */
220  DMAC2.RSAR = 0xFFFFE840; /* 各値のリロード値を設定 */
221
222  /* Configure DRAR
223  b31:0 DMA リロード時のデスティネーションアドレス */
224  DMAC2.RDAR = 0xFFFF82000;
225
226  /* Configure RDMATCR
227  b31:24 Reserved
228  b23:0 DMA リロード時の転送回数 = 1 */
229  DMAC2.RDMATCR = 0x0001;
230
231  /* Configure IPR03
232  b15:12 DMAC0 = H'0 対応する割り込みの優先順位を 0 に設定
233  b11:8 DMAC1 = H'0 対応する割り込みの優先順位を 0 に設定
234  b7:4 DMAC2 = H'F 対応する割り込みの優先順位を 15 に設定
235  b3:0 DMAC3 = H'0 対応する割り込みの優先順位を 0 に設定 */
236  INTC.IPR03.WORD = 0x00F0; /* DMAC2 転送終了割り込みの優先順位を 15 に設定 */
237
238  /* Configure CHFR
239  b7:5 Reserved
240  b4 HE = 0 DMA ハーフエンドフラグをクリア
241  b3:1 Reserved
242  b0 TE = 0 DMA トランスファエンドフラグをクリア */

```

```

243   DMAC2.CHFR.BYTE ^= 0x11; /* DMAC の転送終了フラグをクリア */
244
245   /* Configure CHCR
246   DE = 1 DMA 転送を許可 */
247   DMAC2.CHCR.BIT.DE = 1; /* DMAC チャンネル 2 の動作を許可 */
248
249   /* Configure DMAOR
250   DME = 1 全てのチャンネルの DMA 転送を許可 */
251   DMACCOM.DMAOR.BIT.DME = 1; /* DMAC の動作を許可 */
252   } /* End of function DMA_init() */
253
254   /*****
255   * Function Name : DEI2
256   * Description   : DMAC 転送終了割り込み
257   * Arguments     : none
258   * Return Value  : none
259   *****/
260   void DEI2(void)
261   {
262     /* Configure CHFR
263     b7:5 Reserved
264     b4 HE = 0 DMA ハーフエンドフラグをクリア
265     b3:1 Reserved
266     b0 TE = 0 DMA トランスファエンドフラグをクリア */
267     DMAC2.CHFR.BYTE ^= 0x11; /* DMAC の転送終了フラグをクリア */
268
269     /* Configure TSRG
270     b7:2 Reserved
271     b1 OVFG0 = 0 オーバフローフラグをクリア
272     b0 CMFG0 = 0 コンペアマッチフラグをクリア */
273     ATUG.SUBBLOCK[1].TSRG.BYTE &= 0x00; /* 各ステータスフラグをクリア */
274   } /* End of function DEI2() */

```

2.22.6 SH7253 グループでご使用の場合の変更点

本文 2.22 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください。

2.23 動作例 22 コンペアマッチ割り込み発生[タイマ H]

2.23.1 概要

タイマカウンタ 1H がコンペアマッチを発生する度に、タイマカウンタ 2H がカウントアップを行うことで、タイマカウンタ 1H のコンペアマッチ回数を計測します。

2.23.2 使用機能説明

表 2.24に関連レジスタの機能割り付けを示します。

表 2.24 ATU-III機能割り付け

関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケーラの動作を設定します。
	PSCR0	プリスケーラ 0 の分周比を設定します。
タイマ H レジスタ	TCRH	タイマ H の各機能を設定します。
	OCR1H	タイマカウンタ 1H のコンペアマッチの値を設定します。
	TSRH	オーバーフロー、コンペアマッチが発生した場合にフラグがセットされます。
	TCNT1H	選択したクロックバスでアップカウント動作を行います。
	TCNT2H	TCNT1H と OCR1H のコンペアマッチ発生によってアップカウント動作を行います。
INTC レジスタ	IPR20	CMFH のコンペアマッチ割り込みの優先度を設定します。

2.23.3 動作説明

図 2.38に動作原理を示します。これに従って SH72546 のハードウェアおよびソフトウェア処理によってコンペアマッチ回数の計測動作を発生させます。タイマカウンタ 2H は、タイマカウンタ 1H のコンペアマッチ発生ごとにインクリメントされます。

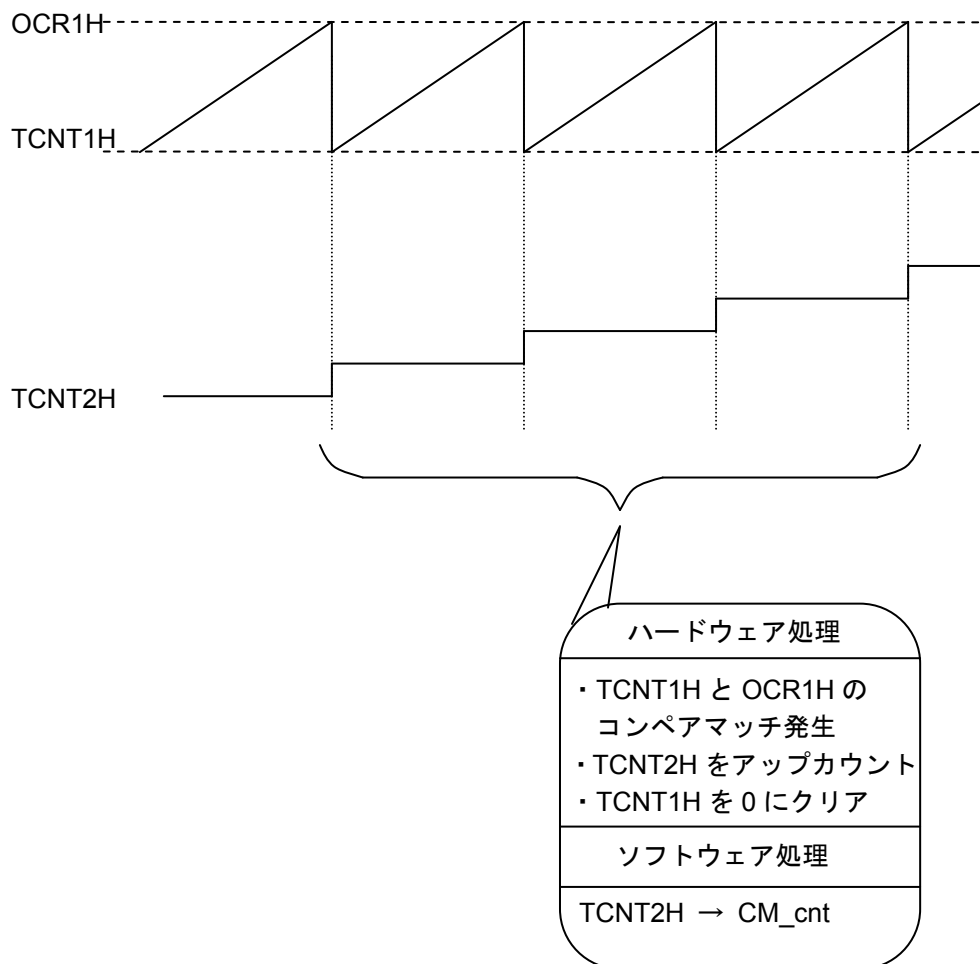


図 2.38 コンペアマッチ回数測定動作原理

2.23.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
コンペアマッチ割り込み	CMIH	TCNT1H と OCR1H のコンペアマッチ発生で割り込み、TCNT2H の値を変数に保存します。

- 使用変数の説明

ラベル名	機能	データ長	使用モジュール名
CM_cnt	コンペアマッチの発生回数を保存します。	unsigned long	コンペアマッチ割り込み

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PSCR0	プリスケアラ 0 の分周比 10 に設定します。	0x0009	メインルーチン
ATUENR	ATU-IIIチャンネルH 及びプリスケアラのカウント動作を許可します。	0x0101	
TCRH	「クロックバス 0 でカウント」「コンペアマッチ時の割り込みを許可」に設定します。	0x01	
IPR20	CMFH による割り込み(CMIH)の優先順位を 15 に設定します。	0x00F0	
OCR1H	TCNT1H とのコンペアマッチの値を設定します。	0x0300	
TSRH	タイマHのコンペアマッチフラグをクリアします。	0x00	コンペアマッチ割り込み

2.23.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Compare-match interrupt occurs.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned long CM_cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ATU-IIIの設定 */
62  /* Configure TCRH
63  b7 Reserved
64  b6:4 CKSELH[2:0] = 000 クロックバス 0
65  b3:1 Reserved
66  b0 CMEH = 1 CMFH による割り込みを許可 */
67  ATUH.TCRH.BYTE = 0x01; /* クロックバス 0 で動作 CMFH による割り込みを許可 */
68
69  /* Configure OCR1H
70  b15:0 OCR1H[15:0] コンペアマッチ値を指定 */
71  ATUH.OCR1H = 0x0300; /* タイマカウンタ 1H のコンペアマッチの値を設定 */
72
73  /* Configure PSCR0
74  b15:10 Reserved
75  b9:0 PSC0[9:0] = H'9 プリスケーラの分周比を 10(設定値は-1)に設定 */
76  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラ 0 の分周比を 10 に設定 */
77
78  /* Configure IPR20
79  b15:12 ATU-G (CMIG0~CMIG3) = H'0 対応する割り込みの優先順位を 0 に設定
80  b11:8 ATU-G (CMIG4、CMIG5) = H'0 対応する割り込みの優先順位を 0 に設定
81  b7:4 ATU-H (CMIH) = H'F 対応する割り込みの優先順位を 15 に設定
82  b3:0 Reserved */
83  INTC.IPR20.WORD = 0x00F0; /* CMFH による割り込み(CMIH)の優先順位を 15 に設定 */
84
85  set_imask(0); /* 割り込みマスケレベルを 0 に設定 */
86
87  /* Configure ATUENR
88  b15:10 Reserved
89  b9 TJE = 0 タイマ J のカウント動作を停止
90  b8 THE = 1 タイマ H のカウント動作を許可
91  b7 TGE = 0 タイマ G のカウント動作を停止
92  b6 TFE = 0 タイマ F のカウント動作を停止
93  b5 TEE = 0 タイマ E のカウント動作を停止
94  b4 TDE = 0 タイマ D のカウント動作を停止
95  b3 TCE = 0 タイマ C のカウント動作を停止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

96     b2 TBE = 0 タイマ B のカウント動作を停止
97     b1 TAE = 0 タイマ A のカウント動作を停止
98     b0 PSCE = 1 プリスケーラのカウント動作を許可 */
99     ATUCTRL.ATUENR.WORD = 0x0101; /* タイマ H およびプリスケーラのカウントをスタート */
100
101     while(1);
102 } /* End of function main() */
103
104 /*****
105  * Function Name : CMIH
106  * Description   : コンペアマッチ割り込み
107  * Arguments    : none
108  * Return Value  : none
109  *****/
110 void CMIH(void)
111 {
112     /* Configure TSRH
113     b7-3 Reserved
114     b2 OVF2H = 0 オーバフローフラグをクリア
115     b1 OVF1H = 0 オーバフローフラグをクリア
116     b0 CMFH = 0 コンペアマッチフラグをクリア */
117     ATUH.TSRH.BYTE &= 0x00; /* コンペアマッチフラグをクリア */
118
119     /* Configure TCNT2H
120     b31-0 TCNT2H[31:0] TCNT1H と OCR1H とのコンペアマッチ発生回数 */
121     CM_cnt = ATUH.TCNT2H;
122 } /* End of function CMIH() */

```

2.23.6 SH7253 グループでご使用の場合の変更点

本文 2.23 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

- (1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください。

2.24 動作例 23 FIFO レジスタへの値の格納[タイマ J]

2.24.1 概要

- 1) 外部からの信号を受信する度にカウンタの値を FIFO レジスタに保存します。
- 2) FIFO レジスタに格納したデータの個数が設定値以上になると、FIFO データフル割り込みで DMAC を起動し、格納された値を転送し、転送した FIFO レジスタに格納されていた値をクリアします。

2.24.2 使用機能説明

表 2.25に関連レジスタの機能割り付けを示します。

表 2.25 ATU-III機能割り付け

関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ J レジスタ	TCRJ0	タイマ J の各機能を設定します。
	FCRJ0	FIFO レジスタの動作を設定します。
	TSTRJ0	タイマ J サブブロックの動作を許可します。
	TIERJ0	データフル時の割り込みを許可します。
	TCNTJ0	選択したクロックバスでアップカウント動作を行います。
	FIFOJ0	コンペアマッチ発生時に TCNTJ0 の値を保存します。
	FDNRJ0	FIFOJ0 に保存されている数の個数が読み出されます。
	OCRJ0	TCNTJ0 のコンペアマッチの値を設定します。
	TSRJ0	タイマ J サブブロック 0 のステータスレジスタです。
DMAC レジスタ	CHCR1	DMAC の各機能の設定をします。
	DMAOR	転送時の各チャネルの動作を設定します。
	DMARS0	DMAC の拡張リソースを設定します。
	SAR1	DMAC 転送の転送先のアドレスを設定します。
	RSAR1	リロード時の転送先アドレス値を設定します。
	DAR1	DMAC 転送の転送元のアドレスを設定します。
	RDAR1	リロード時の転送元アドレス値を設定します。
	DMATCR1	DMAC 転送の転送回数を設定します。
	RDMATCR1	リロード時の転送回数を設定します。
	CHFR1	DMAC の各フラグがセットされます。
PFC レジスタ	PDCCR2	ポート D の端子機能を設定します。
	PECR1	ポート E の端子機能を設定します。
INTC レジスタ	IPR03	DMAC の転送終了割り込みの優先度を設定します。
	IPR21	タイマ J データフル割り込みの優先度を設定します。

2.24.3 動作説明

図 2.39に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってコンペアマッチ回数の計測動作を発生させます。（図は FCRJ0 で設定した基準値が 4 の場合）

DMAC の転送元アドレスを FIFO レジスタに設定することで、FIFO に格納されたカウンタの値を取得します。

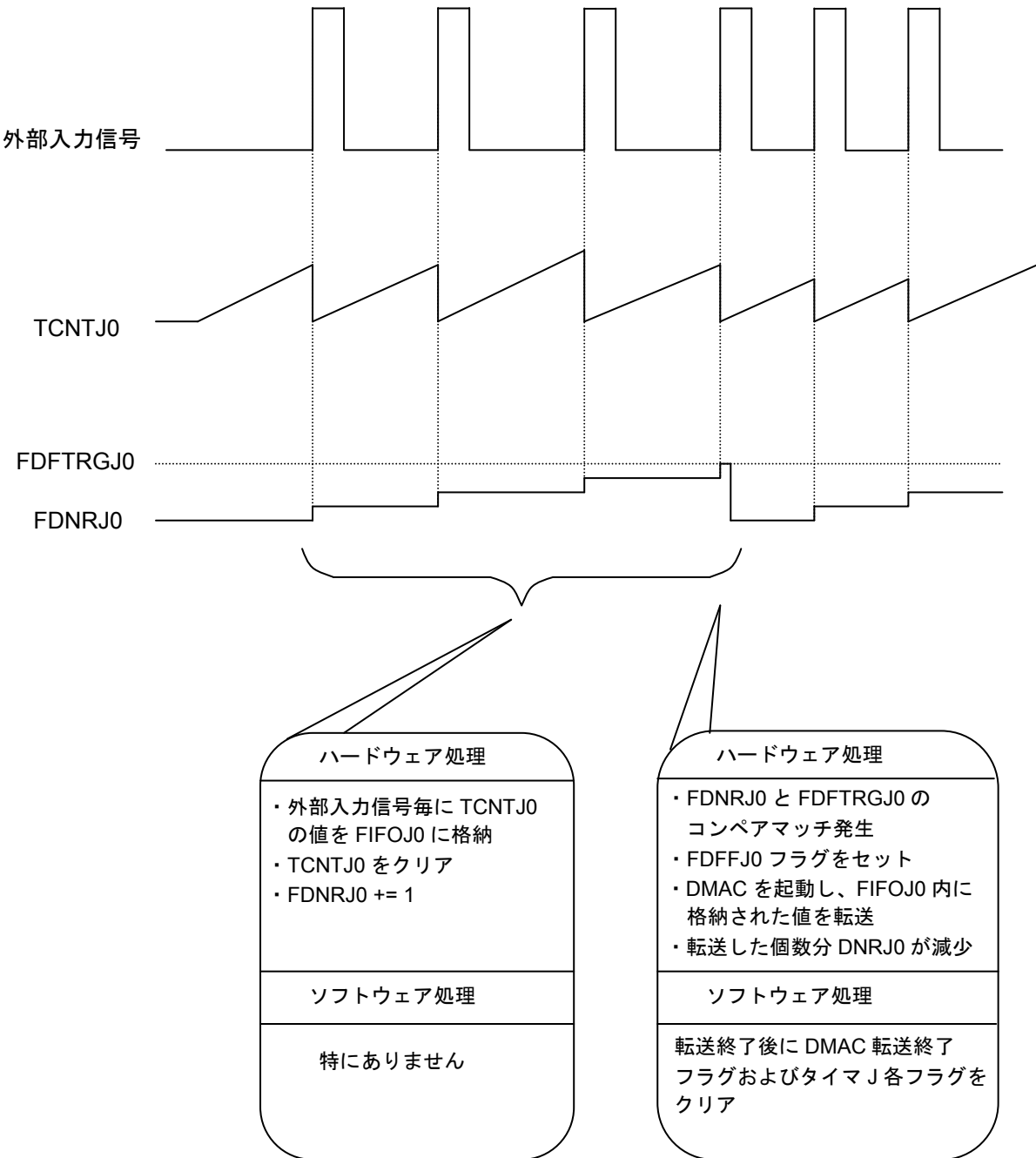


図 2.39 パルスの High 幅計測動作原理

2.24.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
DMAC 設定ルーチン	DMAC_init	TCNTJ0 と OCRJ0 のコンペアマッチ発生で割り込み要求が発生し、TCNT2H の値を変数に保存します。
DMAC 転送終了割り込み	DEI1	DMAC の転送が終了すると割り込み要求が発生し、各フラグのクリア、DMAC の送信先アドレスの増加を行います。

● 使用変数の説明

このタスクでは変数を使用していません。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	メインルーチン
PDCR2	PD12 を TIJ0 入力に設定します。	0x0300	
PECR1	PE6 を TOE00 出力に設定します。	0x1000	
ATUENR	ATU-IIIのチャンネルJ及びプリスケアラのカウント動作を許可します。	0x0201	
TCRJ0	「クロックバス 0 でカウント」「ノイズキャンセルを行わない」「対立ち上がりエッジ」に設定します。	0x01	
FCRJ0	「FIFO レジスタを常に有効」「FIFO レジスタに格納する値の個数を 6 個」に設定します	0x81	
TSRJ0	各ステータスフラグをクリアします。	0x00	
TIERJ0	FIFO データフル時の割り込みを許可します。	0x01	
TSTRJ	タイマ J サブブロック[0]を動作させる。	0x01	
CHCR1	「転送サイズは word 単位」「転送開始リソースは拡張リソースセクタ」「転送方式はバースト転送」「MATCR をリロード」「転送先のアドレスを転送毎に増加」「転送終了割り込みを許可」「このチャンネルの動作を許可」に設定します。	0x9000482D	DMAC 設定ルーチン
DMARS0	DMAC チャンネル 1 の拡張リソースを「ATU-III J0」に設定します。	0xE300	
SAR1	DMA 転送元アドレスを設定します。	0xFFFFFFFF9C	
DAR1	DMA 転送先アドレスを設定します。	0xFFFF80000	
DMATCR1	DMA 転送の回数を設定します。	0x0006	
RSAR1	DMA 転送元アドレスのリロード値を設定します。	0xFFFFFFFF9C	
RDAR1	DMA 転送先アドレスのリロード値を設定します。	0xFFFF80000	
RDMATCR1	DMA 転送の回数のリロード値を設定します	0x006	
CHFR1	DMA 転送に関する各フラグがセットされます。	—	
DMAOR	DMAC 全体の動作の許可を設定します	0x0001	
TSRJ0	各ステータスフラグをクリアします。	0x00	DMAC 転送終了 割り込み
CHFR1	DMAC の転送終了フラグをクリアします。	HE = 0 TE = 0	

2.24.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Store a value into FIFO register.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```



```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50  void DMAC_init(void); /* DMA 設定ルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments    : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PDCR2
63  b15:12 Reserved
64  b11:10 PD13MD[1:0] = 00 PD13 入出力 (ポート)
65  b9:8 PD12MD[1:0] = 11 TIJ0 入力 (ATU-III)
66  b7:6 PD11MD[1:0] = 00 PD11 入出力 (ポート)
67  b5:4 PD10MD[1:0] = 00 PD10 入出力 (ポート)
68  b3:2 PD9MD[1:0] = 00 PD9 入出力 (ポート)
69  b1:0 PD8MD[1:0] = 00 PD8 入出力 (ポート) */
70  PORTD.CR2.WORD = 0x0300; /* PD12 を TIJ0 入力に設定 */
71
72  /* Configure PECR1
73  b15:14 PE7MD[1:0] = 00 PE7 入出力 (ポート)
74  b13:12 PE6MD[1:0] = 01 TOE00 出力 (ATU-III)
75  b11 Reserved
76  b10 PE5MD = 0 PE5 入出力 (ポート)
77  b9 Reserved
78  b8 PE4MD = 0 PE4 入出力 (ポート)
79  b7 Reserved
80  b6 PE3MD = 0 PE3 入出力 (ポート)
81  b5:4 PE2MD[1:0] = 00 PE2 入出力 (ポート)
82  b3:2 PE1MD[1:0] = 00 PE1 入出力 (ポート)
83  b1 Reserved
84  b0 PE0MD = 0 PE0 入出力 (ポート) */
85  PORTE.CR1.WORD = 0x1000; /* PE6 を TOE00 出力に設定 */
86
87  /* ATU-III の設定 */
88  /* Configure TCRJ0
89  b7 Reserved
90  b6:4 CKSELJ0[2:0] = 000 クロックバス 0
91  b3 Reserved
92  b2 NCEJ0 = 0 TIJ0 入力のノイズキャンセラ機能を無効に
93  b1:0 IOJ0[1:0] = 01 TIJ0 の立ち上がりでインプットキャプチャ */

```

```

94  ATUJ.SUBBLOCK[0].TCRJ.BYTE = 0x01; /* タイマサブブロック 0 を「クロックバス 0」「ノイズ
    キャンセル無し」
95      「立ち上がりエッジでキャプチャ」に設定 */
96
97  /* Configure FCRJ0
98  b7 FIFOENJ0 = 1 FIFO レジスタ J0 は有効
99  b6 Reserved
100 b5 FVCRENJ0 = 0 OCRJ0 でのコンペアマッチで FIFO を有効にしない
101 b4 FRSTJ0 = 0 何の動作もしない
102 b3:2 Reserved
103 b1:0 FDFTRGJ0[1:0] = 01 タイマ J ステータスレジスタのフラグセット基準データ数は 6 個 */
104 ATUJ.SUBBLOCK[0].FCRJ.BYTE = 0x81; /* FIFO レジスタを常に有効に、FIFO に格納するデータ
    数を 6 に設定 */
105
106 /* Configure TSRJ0
107 b7:5 Reserved
108 b4 FVLDFJ0 = 0 FIFO レジスタが有効状態でない
109 b3 CMFJ0 = 0 OCRJ0 と TCNTJn のコンペアマッチの検出なし
110 b2 OVFJ0 = 0 TCNTJ0 にオーバフローなし
111 b1 FDOVFJ0 = 0 FIFO 内のデータは正常であることを示す
112 b0 FDFFJ0 = 0 FIFO 内のデータ数が FDFTRGJ の設定値より少ないことを示す */
113 ATUJ.SUBBLOCK[0].TSRJ.BYTE &= 0; /* 各ステータスフラグをクリア */
114
115 /* Configure TIERJ0
116 b7:3 Reserved
117 b2 OVEJ0 = 0 OVEJ0 による割り込みを禁止
118 b1 FDOVEJ0 = 0 FDOVEJ0 による割り込みを禁止
119 b0 FDFEJ0 = 1 FDFEJ0 による割り込みを許可 */
120 ATUJ.SUBBLOCK[0].TIERJ.BYTE = 0x01; /* FIFO データフル時の割り込みを許可 */
121
122 /* Configure TSTRJ
123 b7:2 Reserved
124 b1 STRJ1 = 0 TCNTJn のカウント動作を停止
125 b0 STRJ0 = 1 TCNTJn のカウント動作を許可 */
126 ATUJ.TSTRJ.BYTE = 0x01; /* タイマ J サブブロック 0 を動作 */
127
128 /* Configure PSCR0
129 b15:10 Reserved
130 b9:0 PSCR0[9:0] = H'9 プリスケアラの分周比を 10(設定値は-1)に設定 */
131 ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケアラ 0 の分周比を 10 に設定 */
132
133 DMAC_init(); /* DMAC 設定ルーチンへ */
134
135 /* Configure IPR21
136 b15:12 ATU-J (DFIJ0、DFIJ1) = H'F 対応する割り込みの優先順位を 15 に設定
137 b11:8 ATU-J (OVIJ0、OVIJ1) = H'0 対応する割り込みの優先順位を 0 に設定
138 b7:4 ATU-J (DOVIJ0、DOVIJ1) = H'0 対応する割り込みの優先順位を 0 に設定
139 b3:0 Reserved*/
140 INTC.IPR21.WORD = 0xF000; /* DFIJ0 の割り込み優先順位を 15 に設定 */
141

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

142  set_imask(0); /* 割り込みマスケレベルを 0 に設定 */
143
144  /* Configure ATUENR
145  b15:10 Reserved
146  b9 TJE = 1 タイマ J のカウント動作を許可
147  b8 THE = 0 タイマ H のカウント動作を停止
148  b7 TGE = 0 タイマ G のカウント動作を停止
149  b6 TFE = 0 タイマ F のカウント動作を停止
150  b5 TEE = 0 タイマ E のカウント動作を停止
151  b4 TDE = 0 タイマ D のカウント動作を停止
152  b3 TCE = 0 タイマ C のカウント動作を停止
153  b2 TBE = 0 タイマ B のカウント動作を停止
154  b1 TAE = 0 タイマ A のカウント動作を停止
155  b0 PSCE = 1 プリスケアラのカウント動作を許可 */
156  ATUCTRL.ATUENR.WORD = 0x0201; /* タイマ J およびプリスケアラのカウントをスタート */
157
158  while(1);
159  } /* End of function main() */
160
161  /*****
162  * Function Name : DMAC_init
163  * Description   : DMAC 設定ルーチン (FIFO レジスタに格納したデータの転送)
164  * Arguments    : none
165  * Return Value  : none
166  *****/
167  void DMAC_init(void)
168  {
169  /* Configure CHCR1
170  b31:30 TC[1:0] = 10 1 回の転送要求で DMATCR の設定回数転送
171  b29 Reserved
172  b28 RLD1 = 1 リロード機能 1 は有効 (ON)
173  b27:26 RLD2[1:0] = 00 リロード機能 2 は無効 (OFF)
174  b25 Reserved
175  b24 IFT = 0 Integer→Floating-point 変換機能は無効 (OFF)
176  b23:19 Reserved
177  b18 HIE = 0 DMATCR = (転送前にセットした DMATCR) /2 で、割り込み要求を禁止
178  b17:16 Reserved
179  b15:14 DM[1:0] = 01 デスティネーションアドレスは増加
180  b13:12 SM[1:0] = 00 ソースアドレスは固定 (16 バイト単位転送時は設定禁止)
181  b11:8 RS[3:0] = 1000 DMA 拡張リソースセクタ
182  b7:6 Reserved
183  b5 TB = 1 バーストモード
184  b4:3 TS[1:0] = 01 ワード (2 バイト) 単位
185  b2 IE = 1 割り込み要求を許可
186  b1 Reserved
187  b0 DE = 0 DMA 転送を禁止 */
188  DMAC1.CHCR.LONG = 0x9000482C; /* 転送サイズ:ワード リソース:拡張リソースセクタ
189  MATCR をリロード DMAC 転送終了割り込みを許可
190  バースト転送 転送先のアドレスを増加 */
191

```

```

192  /* Configure DMARS0
193  b15:10 CH1 MID[5:0] = 111000 DMA0 の転送要求元は ATU-III_J0
194  b9:8 CH1 RID[1:0] = 11 同上
195  b7:2 CH0 MID[5:0] = 000000 default
196  b1:0 CH0 RID[1:0] = 00 同上 */
197  DMACCOM.DMARSx[0].DMARS.WORD = 0xE300; /* CH1 の拡張リソースを ATU-III_J0 に設定 */
198
199  /* Configure SAR0
200  b31:0 DMA ソースアドレス */
201  DMAC1.SAR = 0xFFFFF9C; /* 転送元のアドレスを指定(FIFO レジスタ) */
202
203  /* Configure DAR
204  b31:0 DMA デスティネーションアドレス */
205  DMAC1.DAR = 0xFFFF8000; /* 転送先のアドレスを指定 */
206
207  /* Configure DMATCR
208  b31:24 Reserved
209  b23:0 DMA 転送回数 */
210  DMAC1.DMATCR = 0x0006; /* DMA 転送の回数を設定 */
211
212  /* Configure RSAR
213  b31:0 DMA リロード時のソースアドレス */
214  DMAC1.RSAR = 0xFFFFF9C; /* 各アドレスの値のリロード値を設定 */
215
216  /* Configure DRAR
217  b31:0 DMA リロード時のデスティネーションアドレス */
218  DMAC1.RDAR = 0xFFFF8000;
219
220  /* Configure RDMATCR
221  b31:24 Reserved
222  b23:0 DMA リロード時の転送回数 */
223  DMAC1.RDMATCR = 0x0006;
224
225  /* Configure IPR03
226  b15:12 DMAC0 = H'0 対応する割り込みの優先順位を 0 に設定
227  b11:8 DMAC1 = H'F 対応する割り込みの優先順位を 15 に設定
228  b7:4 DMAC2 = H'0 対応する割り込みの優先順位を 0 に設定
229  b3:0 DMAC3 = H'0 対応する割り込みの優先順位を 0 に設定 */
230  INTC.IPR03.WORD = 0x0F00; /* DMAC1 転送終了割り込みの優先順位を 15 に設定 */
231
232  /* Configure CHFR1
233  b7:5 Reserved
234  b4 HE = 0 DMA ハーフエンドフラグをクリア
235  b3:1 Reserved
236  b0 TE = 0 DMA トランスファエンドフラグをクリア */
237  DMAC1.CHFR.BYTE ^= 0x11; /* DMAC の転送終了フラグをクリア */
238
239  /* Configure CHCR
240  b0 DE = 1 DMA 転送を許可 */
241  DMAC1.CHCR.BIT.DE = 1; /* DMA 転送を許可 */

```

```

242
243  /* Configure DMAOR
244  b0 DME = 1 全てのチャネルの DMA 転送を許可 */
245  DMACCOM.DMAOR.BIT.DME = 1; /* DMAC の動作を許可 */
246  } /* End of function DMAC_init() */
247
248  /*****
249  * Function Name : DEI1
250  * Description   : DMAC 転送終了割り込み
251  * Arguments     : none
252  * Return Value  : none
253  *****/
254  void DEI1(void)
255  {
256  /* Configure TSRJ0
257  b7:5 Reserved
258  b4 FVLDFJ0 = 0 FIFO レジスタが有効状態でない
259  b3 CMFJ0 = 0 OCRJ0 と TCNTJ0 のコンペアマッチの検出なし
260  b2 OVFJ0 = 0 TCNTJ0 にオーバフローなし
261  b1 FDOVFJ0 = 0 FIFO 内のデータは正常であることを示す
262  b0 FDFFJ0 = 0 FIFO 内のデータ数が FDFTRGJ の設定値より少ないことを示す */
263  ATUJ.SUBBLOCK[0].TSRJ.BYTE &= 0x00; /* 各ステータスフラグをクリア */
264
265  /* Configure CHFR1
266  b7:5 Reserved
267  b4 HE = 0 DMA ハーフエンドフラグをクリア
268  b3:1 Reserved
269  b0 TE = 0 DMA トランスファエンドフラグをクリア */
270  DMAC1.CHFR.BYTE ^= 0x11; /* DMAC の転送終了フラグをクリア */
271  } /* End of function DEI1() */

```

2.24.6 SH7253 グループでご使用の場合の変更点

本文 2.24 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodefine.h を SH7253 グループ用のものに差替えてください

(2)TIJ0 端子が、SH7253 グループでは、PJ4 端子 (SH7254R グループでは、PD12 端子))に、また、TOE00 端子が PB4(SH7254R グループでは、PE6 端子)に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.24.5 サンプルプログラムリストの 70 行目を

PORTJ.CR1.WORD=0x0300 に変更

SH7254R グループ : PDCR2 を 0x0300 に設定 (変更前)

SH7253 グループ : PJCR1 を 0x0300 に設定 (変更後)

2.24.5 サンプルプログラムリストの 70 行目を

PORTB.CR2A.WORD=0x0001 に変更

SH7254R グループ : PECR1 を 0x1000 に設定 (変更前)

SH7253 グループ : PBCR2A を 0x0001 に設定 (変更後)

2.25 動作例 24 FIFO レジスタへの値の格納[タイマ J]

2.25.1 概要

- 1) 一定期間 (TCNTJ0 と OCRJ0 のコンペアマッチ発生) 毎に外部信号の受信を有効にします。
- 2) 外部信号エッジを受信する度にタイマカウンタの値を FIFO レジスタに保存します。
- 3) FIFO レジスタに格納したデータの個数が設定値以上になると、FIFO データフル割り込みで DMAC を起動し、格納された値を別の場所に保存し、FIFO レジスタをクリアします。また、外部信号の受信を無効に設定し、次のコンペアマッチまで待機します。

2.25.2 使用機能説明

表 2.26に関連レジスタの機能割り付けを示します。

表 2.26 ATU-III機能割り付け

関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
タイマ J レジスタ	TCRJ0	タイマ J の各機能を設定します。
	FCRJ0	FIFO レジスタの動作を設定します。
	TSTRJ	タイマ J 各サブブロックの動作を許可します。
	TIERJ0	データフル時の割り込みを許可します。
	TCNTJ0	選択したクロックバスでアップカウント動作を行います。
	OCRJ0	TCNTJ0 に対するコンペアマッチの値を設定します。
	FIFOJ0	コンペアマッチ発生時に TCNTJ0 の値を保存します。
	FDNRJ0	FIFOJ0 に保存されている数の個数が読み出されます。
	TSRJ0	タイマ J のステータスフラグが格納されます。
DMAC レジスタ	CHCR1	DMAC の各機能の設定をします。
	DMAOR	転送時の各チャネルの動作を設定します。
	DMARS0	DMAC の拡張リソースを設定します。
	SAR1	DMAC 転送の転送先のアドレスを設定します。
	RSAR1	リロード時の DMAC 転送先アドレスを設定します。
	DAR1	DMAC 転送の転送元のアドレスを設定します。
	RDAR1	リロード時の DMAC 転送元アドレスを設定します。
	DMATCR1	DMAC 転送の転送回数を設定します。
	RDMATCR1	リロード時の転送回数を設定します。
	CHFR1	DMAC の各フラグがセットされます。
PFC レジスタ	PDCR2	ポート D の端子機能を設定します。
INTC レジスタ	IPR03	DMAC の転送終了割り込みの優先度を設定します。
	IPR21	タイマ J データフル割り込みの優先度を設定します。

2.25.3 動作説明

図 2.40に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によってコンペアマッチ回数の計測動作を発生させます。（図は FCRJ0 で設定した基準値が 4 の場合）

DMAC の転送元アドレスを FIFO レジスタに設定することで、FIFO に格納されたカウンタの値を取得します。

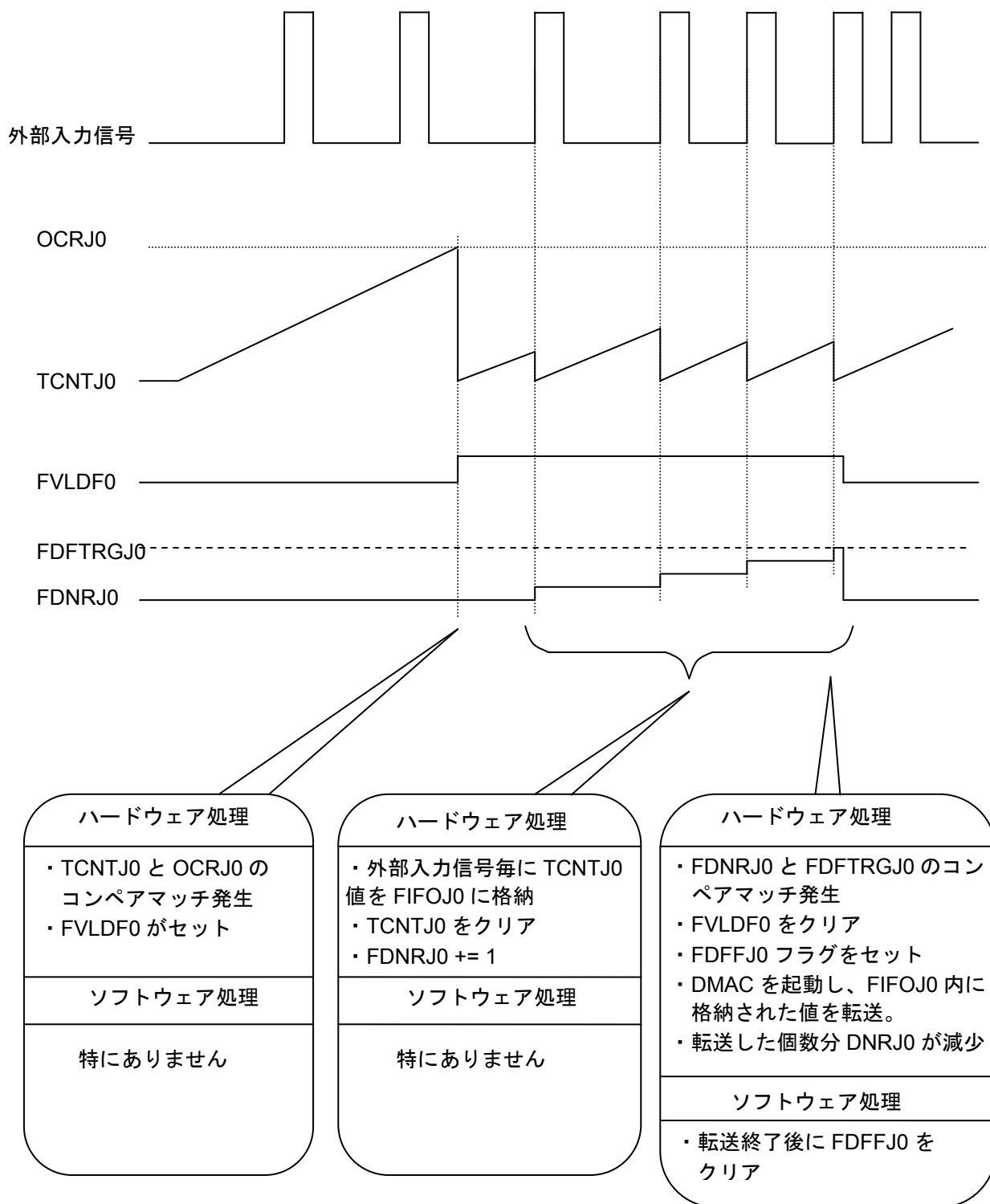


図 2.40 パルスの High 幅計測動作原理

2.25.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
DMAC 設定ルーチン	DMAC_init	TCNTJ0 と OCRJ0 のコンペアマッチで割り込み要求を発生し、TCNT2H の値を変数に保存します。
DMAC 転送終了割り込み	DEI1	DMAC の転送が終了すると割り込み要求を発生し、各フラグのクリア、DMAC の送信先アドレスの増加を行います。

● 使用変数の説明

このタスクでは変数を使用していません。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PDCR2	PD12 を TIJ0 入力に設定します。	0x0300	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-III チャンネル J 及びプリスケアラのカウント動作を許可します。	0x0201	
TCRJ0	「クロックバス 0 でカウント」「ノイズキャンセルを行わない」「対立ち上がりエッジ」に設定します。	0x01	
FCRJ0	「FIFO レジスタをコンペアマッチ後のみ有効」「FIFO に格納する値の個数は 6 個」に設定します。	0x21	
TSRJ0	各ステータスフラグがセットされます。	—	
TCNTJ0	TCRJ で設定したクロックバスでアップカウントを行います。	—	
OCRJ0	TCNTJ0 と対応するコンペアマッチの値を設定します。	0x00C8	
TIERJ0	FIFO データフル時の割り込みを許可します。	0x01	
TSTRJ	タイマ J サブブロック[0]を動作させる。	0x01	
CHCR1	DMAC の各設定を「転送サイズは word 単位」「転送開始リソースは拡張リソースセクタ」「転送方式はバースト転送」「MATCR をリロード」「転送先のアドレスを転送毎に増加」「転送終了割り込みを許可」「このチャンネルの動作を許可」に設定します。	0x9000482D	DMAC 設定ルーチン
DMARS0	DMAC チャンネル 1 の拡張リソースを「ATU-III J0」に設定します。	0xE300	
SAR1	DMA 転送元アドレスを設定します。	0xFFFFFFFF9C	
DAR1	DMA 転送先アドレスを設定します。	0xFFF80000	
DMATCR1	DMA 転送の回数を設定します。	0x0006	
RSAR1	DMA 転送元アドレスのリロード値を設定します。	0xFFFFFFFF9C	
RDAR1	DMA 転送先アドレスのリロード値を設定します。	0xFFF80000	
RDMATCR1	DMA 転送の回数のリロード値を設定します	0x006	
CHFR1	DMA 転送に関する各フラグがセットされます。	—	
DMAOR	DMAC 全体の動作の許可を設定します	0x0001	DMAC 転送終了割り込み
TSRJ0	J のステータスフラグをクリアします。	0x00	
CHFR1	DMAC の転送終了フラグをクリアします。	HE = 0 TE = 0	

2.25.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Store a value into FIFO register.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39 /*****
40 Includes <System Includes> , "Project Includes"
41 *****/
42 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
43 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
44
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50  void DMAC_init(void); /* DMA 設定ルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments    : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PDCR2
63  b15:12 Reserved
64  b11:10 PD13MD[1:0] = 00 PD13 入出力 (ポート)
65  b9:8 PD12MD[1:0] = 11 TIJ0 入力 (ATU-III)
66  b7:6 PD11MD[1:0] = 00 PD11 入出力 (ポート)
67  b5:4 PD10MD[1:0] = 00 PD10 入出力 (ポート)
68  b3:2 PD9MD[1:0] = 00 PD9 入出力 (ポート)
69  b1:0 PD8MD[1:0] = 00 PD8 入出力 (ポート) */
70  PORTD.CR2.WORD = 0x0300; /* PD12 を TIJ0 入力に設定 */
71
72  /* ATU-III の設定 */
73  /* Configure TCRJ0
74  b7 Reserved
75  b6:4 CKSELJ0[2:0] = 000 クロックバス 0
76  b3 Reserved
77  b2 NCEJ0 = 0 TIJ0 入力のノイズキャンセラ機能を無効に
78  b1:0 IOJ0[1:0] = 01 TIJ0 の立ち上がりでインプットキャプチャ */
79  ATUJ.SUBBLOCK[0].TCRJ.BYTE = 0x01; /* タイマサブブロック 0 を「クロックバス 0」
80  「ノイズキャンセル無し」「立ち上がりエッジでキャプチャ」に設定 */
81
82  /* Configure FCRJ0
83  b7 FIFOENJ0 = 0 FIFO レジスタ J0 は無効
84  b6 Reserved
85  b5 FVCRENJ0 = 1 OCRJ0 のコンペアマッチ後、FIFO フルまで FIFO レジスタを有効に
86  b4 FRSTJ0 = 0 何の動作もしない
87  b3:2 Reserved
88  b1:0 FDFTRGJ0[1:0] = 01 FDFJ0 フラグをセットする基準データ数を 6 個に設定 */
89  ATUJ.SUBBLOCK[0].FCRJ.BYTE = 0x21; /* FIFO レジスタをコンペアマッチ発生後から FIFO フ
90  ルまでの間のみに有効に設定
91  FIFO に格納するデータ数を 6 に設定 */
92
93  /* Configure OCRJ0
94  b15:0 OCRJ0[15:0] TCNTJ0 に対するアウトプットコンペア */
95  ATUJ.SUBBLOCK[0].OCRJ = 0x00C8; /* コンペアマッチの値を設定 */

```

```

95
96  /* Configure TSRJ0
97  b7:5 Reserved
98  b4 FVLDFJ0 = 0 FIFO レジスタが有効状態でない
99  b3 CMFJ0 = 0 OCRJ0 と TCNTJ0 のコンペアマッチの検出なし
100 b2 OVFJ0 = 0 TCNTJ0 にオーバフローなし
101 b1 FDOVFJ0 = 0 FIFO 内のデータは正常であることを示す
102 b0 FDFJ0 = 0 FIFO 内のデータ数が FDFTRGJ の設定値より少ないことを示す */
103 ATUJ.SUBBLOCK[0].TSRJ.BYTE &= 0; /* 各ステータスフラグをクリア */
104
105 /* Configure TIERJ0
106 b7:3 Reserved
107 b2 OVEJ0 = 0 OVEJ0 による割り込みを禁止
108 b1 FDOVEJ0 = 0 FDOVEJ0 による割り込みを禁止
109 b0 FDFEJ0 = 1 FDFEJ0 による割り込みを許可 */
110 ATUJ.SUBBLOCK[0].TIERJ.BYTE = 0x01; /* FIFO データフル時の割り込みを許可 */
111
112 /* Configure TSTRJ
113 b7:2 Reserved
114 b1 STRJ1 = 0 TCNTJ1 のカウント動作を停止
115 b0 STRJ0 = 1 TCNTJ0 のカウント動作を許可 */
116 ATUJ.TSTRJ.BYTE = 0x01; /* タイマ J サブブロック 0 を動作 */
117
118 /* Configure PSCR0
119 b15:10 Reserved
120 b9:0 PSC0[9:0] ' H'9 プリスケアラの分周比を 10(設定値は-1)に設定 */
121 ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケアラの分周比を 10 に設定 */
122
123 DMAC_init(); /* DMAC 設定ルーチンへ */
124
125 /* Configure IPR21
126 b15:12 ATU-J (DFIJ0、DFIJ1) ' H'F 対応する割り込みの優先順位を 15 に設定
127 b11:8 ATU-J (OVIJ0、OVIJ1) ' H'0 対応する割り込みの優先順位を 0 に設定
128 b7:4 ATU-J (DOVIJ0、DOVIJ1) ' H'0 対応する割り込みの優先順位を 0 に設定
129 b3:0 Reserved */
130 INTC.IPR21.WORD = 0xF000; /* DFIJ0 の割り込み優先順位を 15 に設定 */
131
132 set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
133
134 /* Configure ATUENR
135 b15:10 Reserved
136 b9 TJE = 1 タイマ J のカウント動作を許可
137 b8 THE = 0 タイマ H のカウント動作を停止
138 b7 TGE = 0 タイマ G のカウント動作を停止
139 b6 TFE = 0 タイマ F のカウント動作を停止
140 b5 TEE = 0 タイマ E のカウント動作を停止
141 b4 TDE = 0 タイマ D のカウント動作を停止
142 b3 TCE = 0 タイマ C のカウント動作を停止
143 b2 TBE = 0 タイマ B のカウント動作を停止
144 b1 TAE = 0 タイマ A のカウント動作を停止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

145  b0 PSCE = 1 プリスケアラのカウント動作を許可 */
146  ATUCTRL.ATUENR.WORD = 0x0201; /* タイマ J およびプリスケアラのカウントをスタート */
147
148  while(1);
149  } /* End of function main() */
150
151  /*****
152  * Function Name : DMAC_init
153  * Description   : DMAC 設定ルーチン (FIFO レジスタに格納したデータ) 転送)
154  * Arguments    : none
155  * Return Value  : none
156  *****/
157  void DMAC_init(void)
158  {
159  /* Configure CHCR1
160  b31:30 TC[1:0] = 10 1 回の転送要求で DMATCR の設定回数転送
161  b29 Reserved
162  b28 RLD1 = 1 リロード機能 1 は有効 (ON)
163  b27:26 RLD2[1:0] = 00 リロード機能 2 は無効 (OFF)
164  b25 Reserved
165  b24 IFT = 0 Integer→Floating-point 変換機能は無効 (OFF)
166  b23:19 Reserved
167  b18 HIE = 0 DMATCR = (転送前にセットした DMATCR) /2 で、割り込み要求を禁止
168  b17:16 Reserved
169  b15:14 DM[1:0] = 01 デスティネーションアドレスは増加
170  b13:12 SM[1:0] = 00 ソースアドレスは固定 (16 バイト単位転送時は設定禁止)
171  b11:8 RS[3:0] = 1000 DMA 拡張リソースセクタ
172  b7:6 Reserved
173  b5 TB = 1 バーストモード
174  b4:3 TS[1:0] = 01 ワード (2 バイト) 単位
175  b2 IE = 1 割り込み要求を許可
176  b1 Reserved
177  b0 DE = 0 DMA 転送を禁止 */
178  DMAC1.CHCR.LONG = 0x9000482C; /* 転送サイズ:ワード リソース:拡張リソースセクタ
179      MATCR をリロード DMAC 転送終了割り込みを許可
180      バースト転送 転送先のアドレスを増加 */
181
182  /* Configure DMARS0
183  b15:10 CH1 MID[5:0] = 111000 DMA0 の転送要求元は ATU-III_J0
184  b9:8 CH1 RID[1:0] = 11 同上
185  b7:2 CH0 MID[5:0] = 000000 default
186  b1:0 CH0 RID[1:0] = 00 同上 */
187  DMACCOM.DMARSx[0].DMARS.WORD = 0xE300; /* CH1 の拡張リソースを ATU-III J0 に設定 */
188
189  /* Configure SAR0
190  b31:0 DMA ソースアドレス */
191  DMAC1.SAR = 0xFFFFF9C; /* 転送元のアドレスを指定(FIFO レジスタ) */
192
193  /* Configure DAR
194  b31:0 DMA デスティネーションアドレス */

```

```

195  DMAC1.DAR = 0xFFFF80000; /* 転送先のアドレスを指定 */
196
197  /* Configure DMATCR
198  b31:24 Reserved
199  b23:0 DMA 転送回数 */
200  DMAC1.DMATCR = 0x0006; /* DMA 転送の回数を設定 */
201
202  /* Configure RSAR
203  b31:0 DMA リロード時のソースアドレス */
204  DMAC1.RSAR = 0xFFFFF9C; /* 各アドレスの値のリロード値を設定 */
205
206  /* Configure DRAR
207  b31:0 DMA リロード時のデスティネーションアドレス */
208  DMAC1.RDAR = 0xFFFF80000;
209
210  /* Configure RDMATCR
211  b31:24 Reserved
212  b23:0 DMA リロード時の転送回数 */
213  DMAC1.RDMATCR = 0x0006;
214
215  /* Configure IPR03
216  b15:12 DMAC0 ' H'0 対応する割り込みの優先順位を 0 に設定
217  b11:8 DMAC1 ' H'F 対応する割り込みの優先順位を 15 に設定
218  b7:4 DMAC2 ' H'0 対応する割り込みの優先順位を 0 に設定
219  b3:0 DMAC3 ' H'0 対応する割り込みの優先順位を 0 に設定 */
220  INTC.IPR03.WORD = 0x0F00; /* DMAC1 転送終了割り込みの優先順位を 15 に設定 */
221
222  /* Configure CHFR1
223  b7:5 Reserved
224  b4 HE = 0 DMA ハーフエンドフラグをクリア
225  b3:1 Reserved
226  b0 TE = 0 DMA トランスファエンドフラグをクリア */
227  DMAC1.CHFR.BYTE ^= 0x11; /* DMAC の転送終了フラグをクリア */
228
229  /* Configure CHCR
230  b0 DE = 1 DMA 転送を許可 */
231  DMAC1.CHCR.BIT.DE = 1; /* DMA 転送を許可 */
232
233  /* Configure DMAOR
234  b0 DME = 1 全てのチャンネルの DMA 転送を許可 */
235  DMACCOM.DMAOR.BIT.DME = 1; /* DMAC の動作を許可 */
236  } /* End of function DMAC_init() */
237
238  /*****
239  * Function Name : DEI1
240  * Description   : DMAC 転送終了割り込み
241  * Arguments     : none
242  * Return Value  : none
243  *****/
244  void DEI1(void)

```

```
245  {
246  /* Configure TSRJ0
247  b7:5 Reserved
248  b4 FVLDFJ0 = 0 FIFO レジスタが有効状態でない
249  b3 CMFJ0 = 0 OCRJ0 と TCNTJ0 のコンペアマッチの検出なし
250  b2 OVFJ0 = 0 TCNTJ0 にオーバフローなし
251  b1 FDOVFJ0 = 0 FIFO 内のデータは正常であることを示す
252  b0 FDFJ0 = 0 FIFO 内のデータ数が FDFTRGJ の設定値より少ないことを示す */
253  ATUJ.SUBBLOCK[0].TSRJ.BYTE &= 0x10; /* 各ステータスフラグをクリア */
254
255  /* Configure CHFR1
256  b7:5 Reserved
257  b4 HE = 0 DMA ハーフエンドフラグをクリア
258  b3:1 Reserved
259  b0 TE = 0 DMA トランスファエンドフラグをクリア */
260  DMAC1.CHFR.BYTE ^= 0x11; /* DMAC の転送終了フラグをクリア */
261  } /* End of function DEI1() */
```

2.25.6 SH7253 グループでご使用の場合の変更点

本文 2.25 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIJ0 端子が、SH7253 グループでは、PJ4 端子 (SH7254R グループでは、PD12 端子) に、割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.25.5 サンプルプログラムリストの 70 行目を

PORTJ.CR1.WORD=0x0300 に変更

SH7254R グループ : PD12 を 0x0300 に設定 (変更前)

SH7253 グループ : PJ4 を 0x0300 に設定 (変更後)

2.26 動作例 25 倍周クロックの生成[タイマ B]

2.26.1 概要

- 1) 図 2.41に示すように外部信号の周期を計測し、それに対する倍周クロックを生成します。
- 2) 倍周クロックの周期は次の式で求められます。

[倍周クロックの周期(ns)=外部入力信号の周期÷パルスインターバルマルチプライヤレジスタ (PIMR)の設定値]

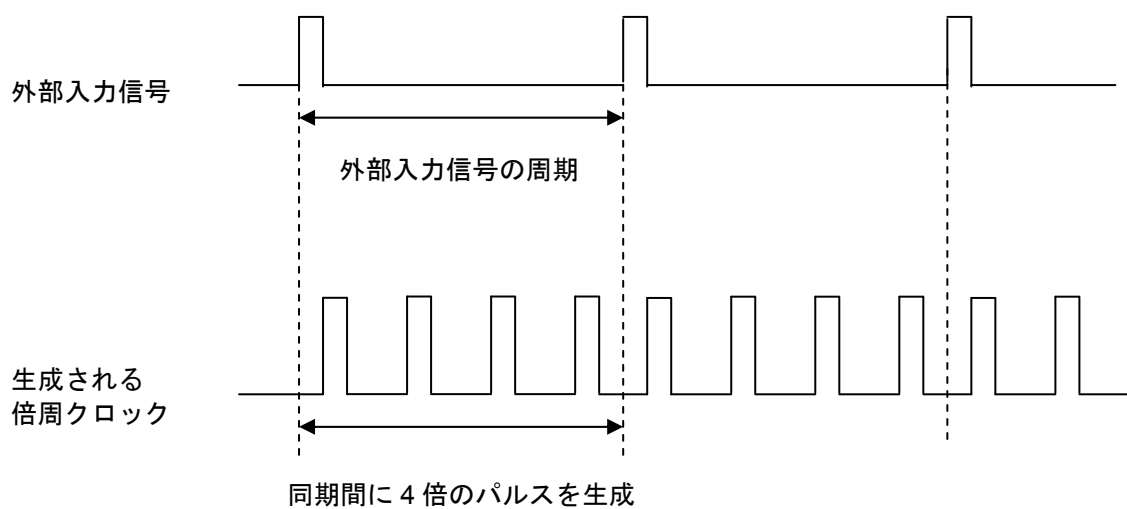


図 2.41 倍周クロックの生成(PIMR=4 の場合)

2.26.2 使用機能説明

表 2.27に使用端子及び関連レジスタの機能割り付けを示します。

表 2.27 ATU-III機能割り付け

使用端子		機能
端子	TIA00	測定するパルスをこの端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	CBCNT	クロックバス 5 のクロックソースを設定します。
タイマ A レジスタ	TCRA	タイマ A の動作クロック、外部入力エッジのイベント出力を設定します。
	TIOR1A	TIA0 の立ち上がりエッジを検出する。
タイマ B レジスタ	TCRB	TCNTB0、TCNTB2 のカウントソースを設定します。
	TIORB	カウント動作の許可等、倍周クロック生成に関する設定を行います。
	PIMR	外部入力信号に対して生成する、倍周クロックの通倍比を設定します。
	TIERB	インプットキャプチャ／コンペアマッチの割り込み要求の許可／禁止を制御します。
	TCNTB3,6	入力エッジ間計測カウンタです。
	OCRB6	TCNTB6 へのコンペアマッチの値を設定します。
タイマ D レジスタ	TCRD	タイマ D の動作を設定します。
	TIOR1D0	タイマ D の入出力を設定します。
	TSRD0	タイマ D のステータスレジスタです。
	OCRD0	TCNT1D0 へのコンペアマッチの値を設定します。
	TIERD0	タイマ D 出力反転の設定を行います。
	TCNT1D0	タイマ D のカウンタの値を設定します。
	DCNTD00 ~02	タイマ D のダウンカウンタの値を設定します。
	DSTRD0	ダウンカウンタの開始/停止を設定します。
PFC レジスタ	PECR1	端子機能を設定します。
INTC レジスタ	IPR07	CMIB6 の割り込み優先順位を設定します。
	IPR10	OCRD00 の割り込み優先順位を設定します。

2.26.3 動作説明

図 2.42に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって入力信号に対する倍周クロックの生成を行います。エッジ間計測ブロックの入力エッジ間計測カウンタ B0 (TCNTB0) からのキャプチャ値に対して、リロード付きのダウンカウントを行います。ダウンカウンタのリロード時に倍週クロック用パルスを生成します。

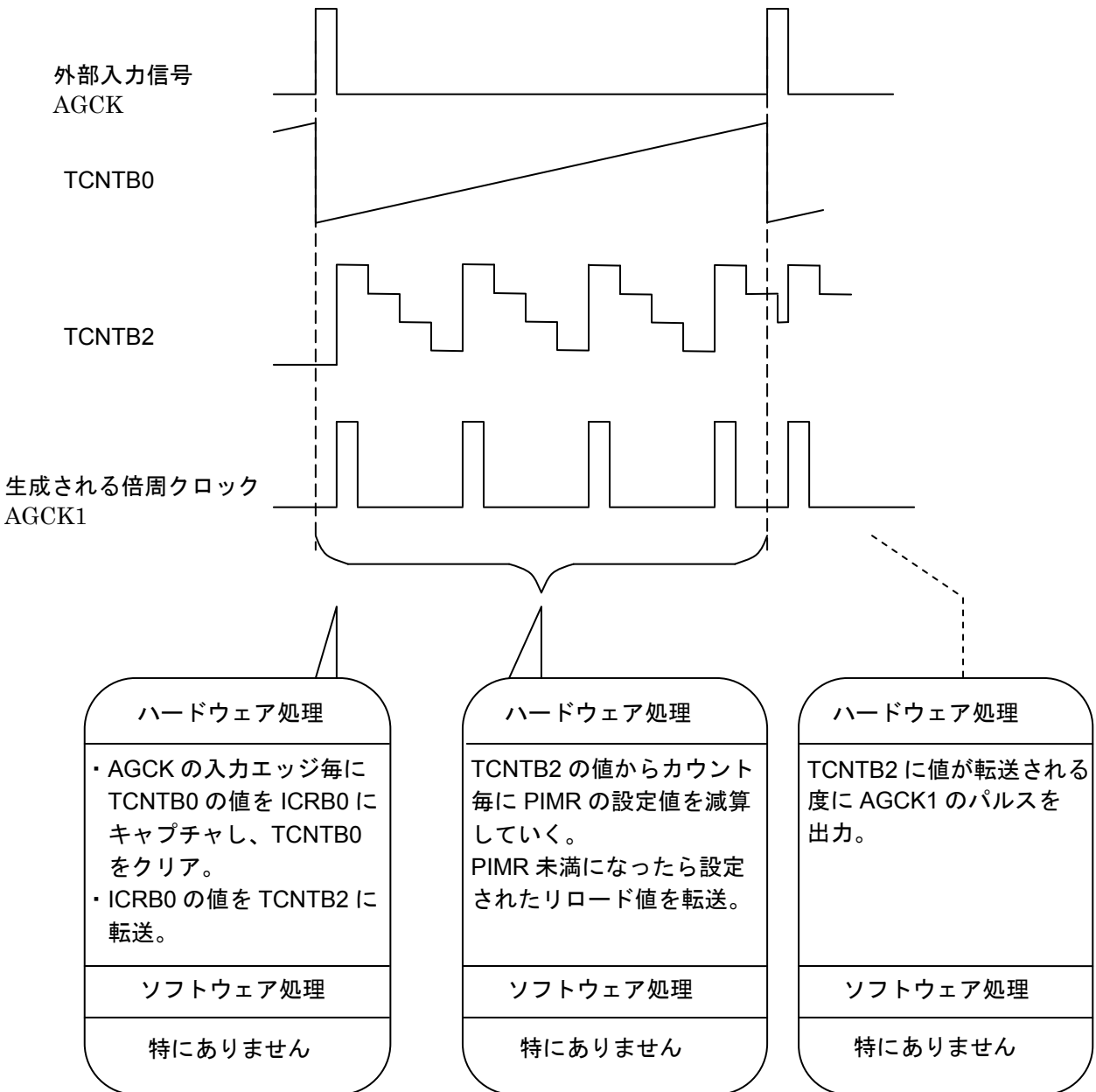


図 2.42 パルス周期計測動作原理(PIMR=4 の場合)

2.26.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。

● 使用変数の説明

ラベル名	機能	データ長	使用モジュール
cnt	ワンショットパルス出力ルーチン参照の回数を記録します。	unsigned char	ワンショットパルス出力ルーチン

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE0 端子を入力端子 TIA0 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIチャネルA,B 及びプリスケアラのカウント動作を許可します。	0x0007	
TCRA	TIA0 への入力信号をイベントとしてタイマ B に出力します。	0x08	
TIOR1A	TIA0 への入力信号の立ち上がりエッジを検出する。	0x0001	
TIORB	TCNTB2 のリロード値を ICRB0 の値に設定、外部イベントの入力を許可します。	0x21	
PIMR	外部入力周期に対する倍周クロックの通倍比を 100 に設定します。	0x0004	
TCRB	TCNTB0, TCNTB2 の使用クロックをクロックバス 0 に設定します。	0x00	
TIERB	TCNTB6, OCRB6 のコンペアマッチ発生タイミングでの割り込み要求を許可します。	0x18	
TCNTB6	TCNTB6 の初期値を 0 に設定します。	0x00000000	
OCRB6	アウトプットコンペア値を 4 に設定します。	0x00002000	
TCRD0	TCNT1D をクロックバス 5 でアップカウントし、DCNTD0 をクロックバス 5 でダウンカウント	0x0055	
TIOR1D	OCRD00 のコンペアマッチ A 発生時、TOD00A 端子からトグル出力します。	0x0103	
TSRD	コンペアマッチ、アンダフローフラグをクリアします。	0x0000	
OCRD	OCRD00 に TCNT1D0 に対応するコンペアマッチ A の条件を 100 に設定します。	0x00006400	
TIERD	OCRD00 のコンペアマッチによる割り込みを許可します。	0x0010	
IPR10	CMID00-CMID03 の割り込み優先順位を 15 に設定します。		
IPR07	CMIB6 のコンペアマッチ割り込み優先順位を 15 に設定します。	0x0F00	
ATUENR	タイマ A,B,D およびプリスケアラのカウントをスタートします。	0x0017	

SH7253 SH7254R グループ

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

TCNTB3	入力エッジ間計測カウント値を 0 に設定します。	0x00000000	ワンショットパルス出カルーチン
TCNT1D0	タイマカウンタ値を 0 に設定します。	0x00000000	
DCNTD00～ DCNTD02	ダウンカウンタ値を設定します。	0x00FA<<8	
DSTRD0	ダウンカウンタ 00 を開始します。 ダウンカウンタ 01 を開始します。 ダウンカウンタ 02 を開始します。	0x01 0x02 0x04	

2.26.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Generate multiple clock.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  unsigned char cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments    : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PECR1
63  b15:14 PE7MD = 0 PE7 入出力 (ポート)
64  b13:12 PE6MD = 0 PE6 入出力 (ポート)
65  b11 Reserved
66  b10 PE5MD = 0 PE5 入出力 (ポート)
67  b9 Reserved
68  b8 PE4MD = 0 PE4 入出力 (ポート)
69  b7 Reserved
70  b6 PE3MD = 0 PE3 入出力 (ポート)
71  b5:4 PE2MD = 0 PE2 入出力 (ポート)
72  b3:2 PE1MD = 0 PE1 入出力 (ポート)
73  b1 Reserved
74  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
75  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
76
77  /* ATU-III(A)の設定 */
78  /* Configure TCRA
79  b7 EVOSEL2A = 0 イベント出力 2A セレクト
80  b6 EVOSEL2B = 0 イベント出力 2B セレクト
81  b5:3 EVOSEL1 = 001 タイマ B へ TIA00 の入力エッジを出力
82  b2:0 CKSELA = 000 カウントクロックにクロックバス 0(プリスケアラ 0)を選択 */
83  ATUA.TCRA.BYTE = 0x08; /* タイマ A をクロックバス 0 のカウントクロックで動作
84     TIA00 の入力をイベントとしてタイマ B に出力 */
85  /* Configure TIOR1A
86  b15:12 Reserved
87  b11:10 IOA5 = 00 TIA のインプットキャプチャ禁止
88  b9:8 IOA4 = 00 TIA のインプットキャプチャ禁止
89  b7:6 IOA3 = 00 TIA のインプットキャプチャ禁止
90  b5:4 IOA2 = 00 TIA のインプットキャプチャ禁止
91  b3:2 IOA1 = 00 TIA のインプットキャプチャ禁止
92  b1:0 IOA0 = 01 TIA の立ち上がりで ICRA にキャプチャ */
93  ATUA.TIOR1A.WORD = 0x0001; /* TIA00 入力の立ち上がりエッジで ICRA にインプットキャプチャ
94  */

```

```

95  /* ATU-III(B)の設定 */
96  /* Configure TIORB
97  b7 LDSEL = 0 TCNTB2 にロードする値に ICRB0 を用いる
98  b6 CTCNTB5 = 0 TCNTB5 のカウントを許可
99  b5 EVCNTB = 1 外部イベントの入力を許可
100 b4 LDEN = 0 リロードレジスタの更新を許可
101 b3 CCS = 0 TCNTB=TCNTB4 のとき、TCNTB4 のカウント動作を停止しない
102 b2:1 Reserved
103 b0 IOB6 = 1 TCNTB6 と OCB6 とのコンペアマッチを許可 */
104 ATUB.TIORB.BYTE = 0x21;
105
106 /* Configure PIMR
107 b15:12 Reserved
108 b11:0 PIM = H'0004 クロックの通倍比を 4 倍に設定 */
109 ATUB.PIMR.WORD = 0x0004; /* 倍周クロックの通倍比を 4 倍に設定 */
110
111 /* Configure TCRB
112 b7:2 Reserved
113 b1:0 CKSELB = 00 カウントソースにクロックバス 0 を選択 */
114 ATUB.TCRB.BYTE = 0x00; /* TCNTB0, TCNTB2 をクロックバス 0 のカウントクロックで動作 */
115
116 /* Configure TIERB
117 b7:6 Reserved
118 b5:4 IREG = 01 CMFB6 有効後、次の外部イベント入力タイミングで割り込み要求を出力
119 b3 CMEB6 = 1 コンペアマッチ割り込み B6 要求の出力を許可
120 b2 CMEB1 = 0 コンペアマッチ割り込み B1 要求の出力を禁止
121 b1 ICEB0 = 0 インプットキャプチャ割り込み B0 要求の出力を禁止
122 b0 CMEB0 = 0 コンペアマッチ割り込み B0 要求の出力を禁止 */
123 ATUB.TIERB.BYTE = 0x18; /* TCNTB6, OCB6 のコンペアマッチ発生のタイミングでの割り込み要求を許可 */
124
125 /* Configure TCNTB
126 b31:0 CNTB0 = 0x00000000 入力エッジ間計測カウント動作数を 0 クリア */
127 ATUB.TCNTB6.LONG = 0x00000000; /* TCNTB6 の初期値を 0 に設定 */
128
129 /* Configure OCB6
130 b31:12 OCB6 = H'00002 アウトプットコンペア値を、2 に設定
131 b11:0 Reserved */
132 ATUB.OCRB6.LONG = 0x00002000; /* TCNTB6 に対応するコンペアマッチの値を 4 に設定 */
133
134 /* タイマ D に関する設定 */
135 /* Configure TCRD
136 b15 Reserved
137 b14 OBREDn = 0 OSBRDn のインプットキャプチャ動作を禁止
138 b13 C2CEDn = 0 タイマ B からの TCNT2Dn カウンタ値クリア要求を禁止
139 b12 C1CEDn = 0 タイマ B からの TCNT1Dn カウンタ値クリア要求を禁止
140 b11 Reserved
141 b10:8 CKSEL2Dn = 000 クロックバス 0 で TCNT2Dn をアップカウント
142 b7 Reserved
143 b6:4 CKSEL1Dn = 101 クロックバス 5 で TCNT1Dn をアップカウント
144 b3 Reserved

```


アドバンスタイマユニット-III (ATU-III) を使用したタイマ動作例

```

145  b2:0 DCSELDn = 101 クロックバス 5 で DCNTDnm をダウンカウント */
146  ATUD.SUBBLOCKA[0].TCRD.WORD = 0x0055; /* TCNT1Dn をクロックバス 5 でアップカウント
147      DCNTD0n をクロックバス 5 でダウンカウント */
148  /* Configure TIOR1D
149  b15:14 OSSDn3 = 00 TODn3A 端子出力なし
150  b13:12 OSSDn2 = 00 TODn2A 端子出力なし
151  b11:10 OSSDn1 = 00 TODn1A 端子出力なし
152  b9:8 OSSDn0 = 01 TODn0A 端子出力なし
153  b7:6 IOADn3 = 00 OCRDn3 コンペアマッチ禁止
154  b5:4 IOADn2 = 00 OCRDn2 コンペアマッチ禁止
155  b3:2 IOADn1 = 00 OCRDn1 コンペアマッチ禁止
156  b1:0 IOADn0 = 11 OCRDn0 コンペアマッチ時、トグル出力 */
157  ATUD.SUBBLOCKB[0].TIOR1D.WORD = 0x0103; /* OCRD00 のコンペアマッチ A 発生時、TOD00A
      端子からトグル出力 */
158
159  /* Configure TSRD
160  b15:14 Reserved
161  b13 OVF2Dn = 0 オーバフローフラグをクリア
162  b12 OVF1Dn = 0 オーバフローフラグをクリア
163  b11 UDFDn3 = 0 アンダーフローフラグをクリア
164  b10 UDFDn2 = 0 アンダーフローフラグをクリア
165  b9 UDFDn1 = 0 アンダーフローフラグをクリア
166  b8 UDFDn0 = 0 アンダーフローフラグをクリア
167  b7 CMFADn3 = 0 コンペアマッチフラグをクリア
168  b6 CMFADn2 = 0 コンペアマッチフラグをクリア
169  b5 CMFADn1 = 0 コンペアマッチフラグをクリア
170  b4 CMFADn0 = 0 コンペアマッチフラグをクリア
171  b3 CMFBDn3 = 0 コンペアマッチフラグをクリア
172  b2 CMFBDn2 = 0 コンペアマッチフラグをクリア
173  b1 CMFBDn1 = 0 コンペアマッチフラグをクリア
174  b0 CMFBDn0 = 0 コンペアマッチフラグをクリア */
175  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンダフローフラグをクリア */
176
177  /* Configure OCRD
178  b31:8 アウトプットコンペアレジスタを H'64(D'100) に設定
179  b7:0 Reserved */
180  ATUD.SUBBLOCKB[0].OCRD[0] = 0x00006400; /* OCRD00 に TCNT1D0 に対応するコンペアマッ
      チ A の条件を 100 に設定 */
181
182  /* Configure TIERD
183  b15:14 Reserved
184  b13 OVE2Dn = 0 OVF2Dn による割り込み要求を禁止
185  b12 OVE1Dn = 0 OVF1Dn による割り込み要求を禁止
186  b11 UDEDn3 = 0 UDFDn3 による割り込み要求を禁止
187  b10 UDEDn2 = 0 UDFDn2 による割り込み要求を禁止
188  b9 UDEDn1 = 0 UDFDn1 による割り込み要求を禁止
189  b8 UDEDn0 = 0 UDFDn0 による割り込み要求を禁止
190  b7 CMEADn3 = 0 CMFADn3 による割り込み要求を禁止
191  b6 CMEADn2 = 0 CMFADn2 による割り込み要求を禁止
192  b5 CMEADn1 = 0 CMFADn1 による割り込み要求を禁止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

193  b4 CMEADn0 = 1 CMFADn0 による割り込み要求を許可
194  b3 CMEBDn3 = 0 CMFBDn3 による割り込み要求を禁止
195  b2 CMEBDn2 = 0 CMFBDn2 による割り込み要求を禁止
196  b1 CMEBDn1 = 0 CMFBDn1 による割り込み要求を禁止
197  b0 CMEBDn0 = 0 CMFBDn0 による割り込み要求を禁止 */
198  ATUD.SUBBLOCKB[0].TIERD.WORD = 0x0010; /* OCRD00 のコンペアマッチによる割り込みを許可
199  */
200  /* Configure IPR10
201  b15:12 ATU-D0(CMID00-CMID03) = H'F 対応する割り込みの優先順位を 15 に設定
202  b11:8 ATU-D0(OVI1D0-OVI2D0) = H'0 対応する割り込みの優先順位を 0 に設定
203  b7:4 ATU-D0(UDID00-UDID03) = H'0 対応する割り込みの優先順位を 0 に設定
204  b3:0 ATU-D1(CMID10-CMID13) = H'0 対応する割り込みの優先順位を 0 に設定 */
205  INTC.IPR10.WORD = 0xF000; /* OCRD00 の割り込み優先順位を 15 に設定 */
206
207  /* ATU-III(共通)の設定 */
208  /* Configure PSCR0
209  b15:10 Reserved
210  b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1) に設定 */
211  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラ 0 の分周比を 10 に設定 */
212
213  /* Configure IPR07
214  b15:12 ATU-B(CMIB0,CMIB1) = H'0 対応する割り込みの優先順位を 0 に設定
215  b11:8 ATU-B(CMIB6,ICIB0) = H'F 対応する割り込みの優先順位を 15 に設定
216  b7:4 ATU-C0(IMIC0-IMIC3) = H'0 対応する割り込みの優先順位を 0 に設定
217  b3:0 ATU-C0(OVIC0) = H'0 対応する割り込みの優先順位を 0 に設定 */
218  INTC.IPR07.WORD = 0x0F00; /* TCNTB6 のコンペアマッチの割り込み優先順位を 15 に設定 */
219
220  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
221
222  /* Configure ATUENR
223  b15:10 Reserved
224  b9 TJE = 0 タイマ J のカウント動作を停止
225  b8 THE = 0 タイマ H のカウント動作を停止
226  b7 TGE = 0 タイマ G のカウント動作を停止
227  b6 TFE = 0 タイマ F のカウント動作を停止
228  b5 TEE = 0 タイマ E のカウント動作を停止
229  b4 TDE = 1 タイマ D のカウント動作を許可
230  b3 TCE = 0 タイマ C のカウント動作を停止
231  b2 TBE = 1 タイマ B のカウント動作を許可
232  b1 TAE = 1 タイマ A のカウント動作を許可
233  b0 PSCE = 1 プリスケーラのカウント動作を許可 */
234  ATUCTRL.ATUENR.WORD = 0x0017; /* タイマ A,B,D およびプリスケーラのカウントをスタート */
235  while(1);
236  } /* End of function main() */
237
238  /*****
239  * Function Name : CMIB6
240  * Description   : コンペアマッチ割り込み
241  * Arguments     : none

```

```

242  * Return Value : none
243  *****/
244  void CMIB6(void)
245  {
246  /* Configure TSRB
247  b7:4 Reserved
248  b3 CMFB6 = 0 コンペアマッチフラグをクリア
249  b2 CMFB1 &= 1 コンペアマッチフラグを維持
250  b1 ICFB0 &= 1 コンペアマッチフラグを維持
251  b0 CMFB0 &= 1 コンペアマッチフラグを維持 */
252  ATUB.TSRB.BYTE &= 0xF7; /* コンペアマッチフラグをクリア */
253  } /* End of function CMIB6() */
254
255  /*****/
256  * Function Name : CMID00
257  * Description : ワンショットパルス出力ルーチン
258  * Arguments : none
259  * Return Value : none
260  *****/
261  void CMID00(void)
262  {
263  /* Configure TSRD
264  b15:14 Reserved
265  b13 OVFDn = 0 オーバフローフラグをクリア
266  b12 OVFDn = 0 オーバフローフラグをクリア
267  b11 UDFDn3 = 0 アンダーフローフラグをクリア
268  b10 UDFDn2 = 0 アンダーフローフラグをクリア
269  b9 UDFDn1 = 0 アンダーフローフラグをクリア
270  b8 UDFDn0 = 0 アンダーフローフラグをクリア
271  b7 CMFADn3 = 0 コンペアマッチフラグをクリア
272  b6 CMFADn2 = 0 コンペアマッチフラグをクリア
273  b5 CMFADn1 = 0 コンペアマッチフラグをクリア
274  b4 CMFADn0 = 0 コンペアマッチフラグをクリア
275  b3 CMFBDn3 = 0 コンペアマッチフラグをクリア
276  b2 CMFBDn2 = 0 コンペアマッチフラグをクリア
277  b1 CMFBDn1 = 0 コンペアマッチフラグをクリア
278  b0 CMFBDn0 = 0 コンペアマッチフラグをクリア */
279  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンダフローフラグをクリア */
280
281  /* Configure TCNTB
282  b31:0 入力エッジ間計測カウンタ値 = 0 */
283  ATUB.TCNTB3.LONG = 0x00000000;
284
285  /* Configure TCNA1D
286  b31:8 タイマカウンタ値 = 0
287  b7:0 Reserved */
288  ATUD.SUBBLOCKA[0].TCNT1D = 0x00000000;
289
290  cnt++;
291  if(cnt == 1)

```

```
292 {
293 /* Configure DCNTD00
294 b31:8 タイマダウンカウンタ値
295 b7:0 Reserved */
296 ATUD.SUBBLOCKB[0].DCNTD[0] = 0x00FA << 8; /* ダウンカウンタ 00 を 250 にセット */
297
298 /* Configure DSTRD0
299 b7:4 Reserved
300 b3 DSTD03=0 DCNTD03 ダウンカウント停止
301 b2 DSTD02=0 DCNTD02 ダウンカウント停止
302 b1 DSTD01=0 DCNTD01 ダウンカウント停止
303 b0 DSTD00=1 DCNTD00 ダウンカウント開始 */
304 ATUD.SUBBLOCKB[0].DSTRD.BYTE = 0x01; /* ダウンカウンタ 00 のダウンカウントをスタート */
305 }
306 else if(cnt == 4)
307 {
308 /* Configure DCNTD01
309 b31:8 タイマダウンカウンタ値
310 b7:0 Reserved */
311 ATUD.SUBBLOCKB[0].DCNTD[1] = 0x00FA<<8; /* ダウンカウンタ 01 を 250 にセット */
312
313 /* Configure DSTRD0
314 b7:4 Reserved
315 b3 DSTD03=0 DCNTD03 ダウンカウント停止
316 b2 DSTD02=0 DCNTD02 ダウンカウント停止
317 b1 DSTD01=0 DCNTD01 ダウンカウント開始
318 b0 DSTD00=1 DCNTD00 ダウンカウント停止 */
319 ATUD.SUBBLOCKB[0].DSTRD.BYTE = 0x02; /* ダウンカウンタ 01 のダウンカウントをスタート */
320 }
321 else if(cnt == 8)
322 {
323 /* Configure DCNTD02
324 b31:8 タイマダウンカウンタ値
325 b7:0 Reserved */
326 ATUD.SUBBLOCKB[0].DCNTD[2] = 0x00FA << 8; /* ダウンカウンタ 02 を 250 にセット */
327
328 /* Configure DSTRD0
329 b7:4 Reserved
330 b3 DSTD03=0 DCNTD03 ダウンカウント停止
331 b2 DSTD02=0 DCNTD02 ダウンカウント開始
332 b1 DSTD01=0 DCNTD01 ダウンカウント停止
333 b0 DSTD00=1 DCNTD00 ダウンカウント停止 */
334 ATUD.SUBBLOCKB[0].DSTRD.BYTE = 0x04; /* ダウンカウンタ 02 のダウンカウントをスタート */
335 }
336 else if(cnt == 10)
337 {
338 cnt = 0;
339 }
340 } /* End of function CMID00() */
```

2.26.6 SH7253 グループでご使用の場合の変更点

本文 2.26 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.26.5 サンプルプログラムリストの 75 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.27 動作例 26 倍周クロックを用いた入力信号の欠け歯の検出[タイマ B]

2.27.1 概要

- 1) 図 2.43に示すように外部信号の周期を計測し、それに対する倍周クロックを生成します。
- 2) 生成した倍周クロックのカウントから、外部入力信号の欠け歯を検出します。

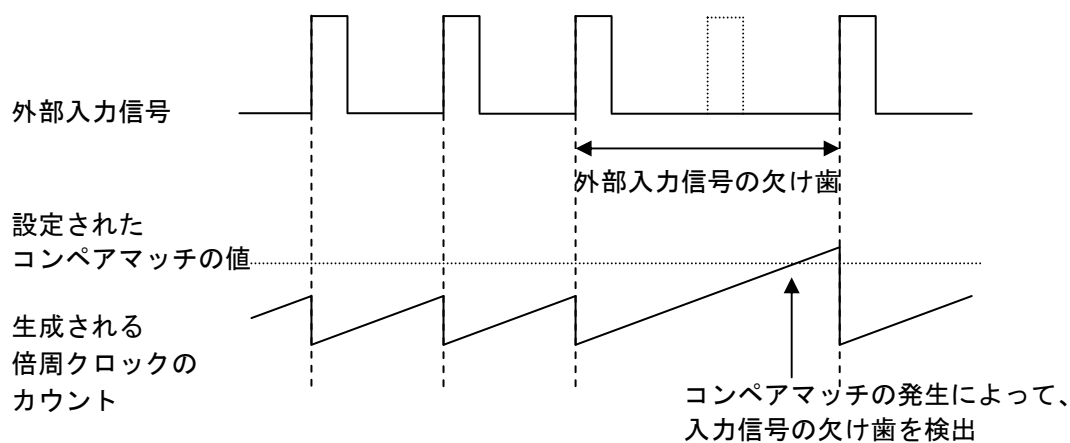


図 2.43 倍周クロックの欠け歯の検出

2.27.2 使用機能説明

表 2.28に使用端子及び関連レジスタの機能割り付けを示します。

表 2.28 ATU-III機能割り付け

使用端子		機能
端子	TIA00	測定するパルスを端子に入力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	CBCNT	クロックバス 5 のクロックソースを設定します。
タイマ A レジスタ	TCRA	タイマ A の動作クロック、外部入力エッジのイベント出力を設定します。
	TIOR1A	TIA00 の立ち上がりエッジを検出する。
タイマ B レジスタ	TCRB	TCNTB0、TCNTB2 のカウントソースを設定します。
	TIORB	カウント動作の許可等、倍周クロック生成に関する設定を行います。
	PIMR	外部入力信号に対して生成する、倍周クロックの通倍比を設定します。
	TIERB	割り込み要求の許可を設定します。
	TCNTB6	生成した倍周クロックでカウント動作を行います。
	TCCLRB	補正カウンタクリア値を設定します。
	OCRB6	欠け歯として検出する入力エッジ間隔を設定します。
	TSRB	タイマ B のステータスが格納されます。
PFC レジスタ	PECR1	使用端子の機能を設定します。
INTC レジスタ	IPR07	CMIB6 のコンペアマッチの優先度を設定します。

2.27.3 動作説明

図 2.44に動作原理を示します。これに従って SH72546R のハードウェアおよびソフトウェア処理によって入力信号の欠け歯の検出を行います。コンペアマッチが発生した時点で、外部クロックに欠け歯があることが判明します。

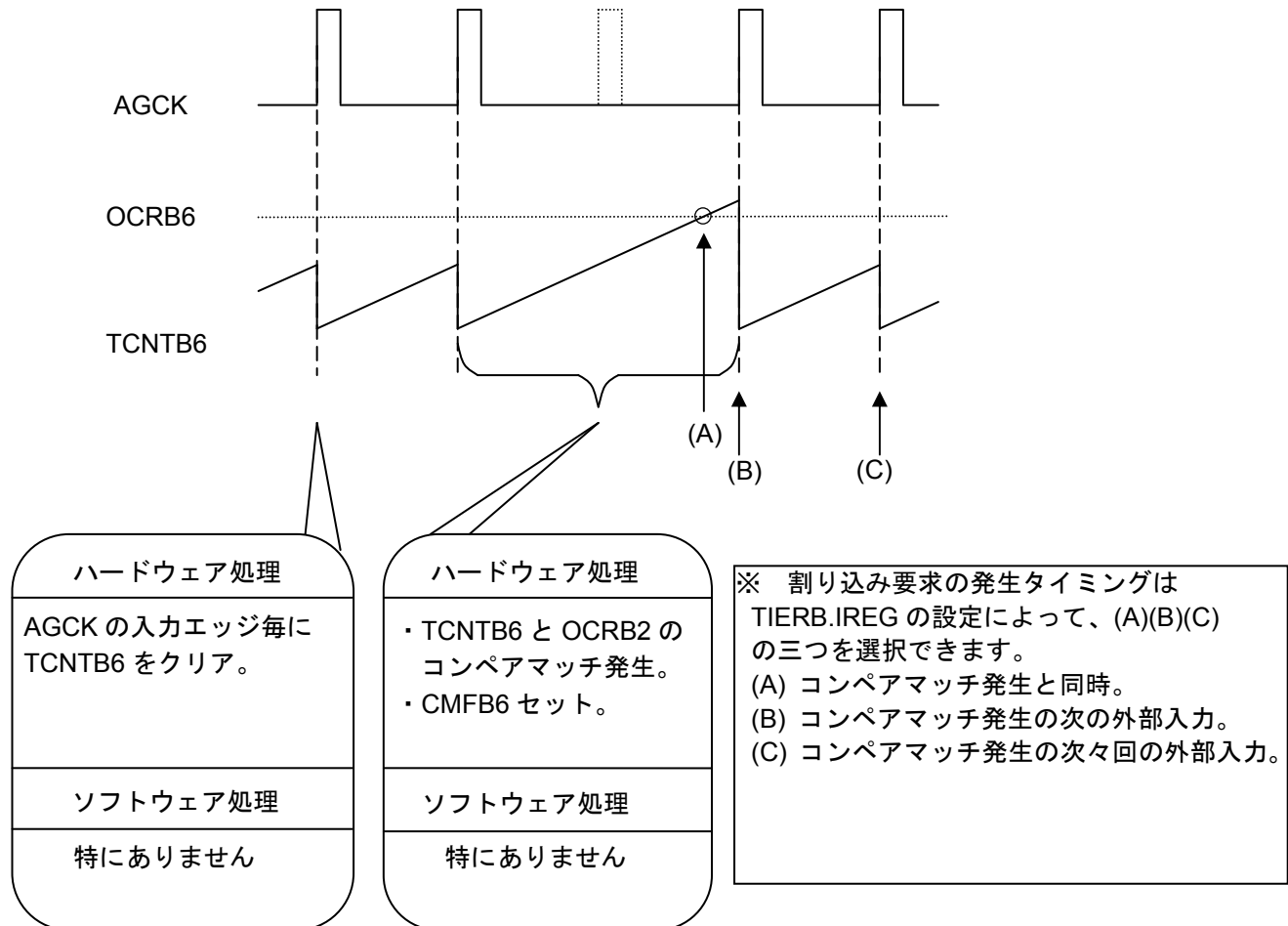


図 2.44 欠け歯検出動作原理

2.27.4 ソフトウェア説明

● モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
欠け歯検出ルーチン	CMIB6	欠け歯を検出するとこのルーチンにジャンプします。

● 使用変数の説明

本タスクでは変数は使用していません。

● 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE0 端子を入力端子 TIA0 に設定します。	0x0001	メインルーチン
PSCR0	プリスケアラ 0 の分周比を 10 に設定します。	0x0009	
ATUENR	ATU-IIIのチャネル A,B 及びプリスケアラのカウント動作を許可します。	0x0007	
CBCNT	クロックバス 5 のクロックソースをタイマ B の倍周補正クロックに設定します。	0x04	
TCRA	TIA0 への入力信号をイベントとしてタイマ B に出力します。	0x08	
TIOR1A	TIA0 への入力信号の立ち上がりエッジを検出する。	0x0001	
TIORB	TCNTB2 のリロード値を ICRB0 の値に設定、外部イベントの入力を許可します。	0x21	
PIMR	外部入力周期に対する倍周クロックの通倍比を 4 に設定します。	0x0004	
TCRB	TCNTB0,TCNTB2 の使用クロックをクロックバス 0 に設定します。	0x00	
TIERB	OCRB6 コンペアマッチ時の割り込みを許可、また割り込み出力のタイミングを設定します。	0x18	
TCNTB6	入力エッジ間計測カウント TCNTB6 の初期値を 0 に設定します。	0x00000000	
OCRB6	欠け歯検出を判断する、TCNTB5 とのコンペアマッチの値を 2 に設定します。※	0x00002000	
TCNTB5	入力エッジ間計測カウント TCNTB5 の初期値を 0 に設定します。	0x00000000	
TCCLRB	TCNTB5 の上限を 150 に設定します。	0x00096000	
CBCNT	クロックバス 5 をタイマ B からの倍周補正クロックに設定	0x04	
IPR07	TCNTB6 のコンペアマッチの割り込み優先順位を 15 に設定します。	0x0F00	
TSRB	コンペアマッチフラグをクリアします。	0x00	コンペアマッチ割り込みルーチン
TCNTB3	入力エッジ間計測カウント TCNTB3 をクリアします。	0x00000000	

※ サンプルプログラムは 1 歯欠けから検出するように値を設定しています。

2.27.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Detection of missing teeth using multiple clock input signal.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Private global variables and functions
48  *****/
49  void main(void); /* メインルーチン */
50
51  /*****
52  * Function Name : main
53  * Description   : The main loop
54  * Arguments     : none
55  * Return Value  : none
56  *****/
57
58  void main(void)
59  {
60  /* ポートの設定 */
61  /* Configure PECCR1
62  b15:14 PE7MD = 0 PE7 入出力 (ポート)
63  b13:12 PE6MD = 0 PE6 入出力 (ポート)
64  b11 Reserved
65  b10 PE5MD = 0 PE5 入出力 (ポート)
66  b9 Reserved
67  b8 PE4MD = 0 PE4 入出力 (ポート)
68  b7 Reserved
69  b6 PE3MD = 0 PE3 入出力 (ポート)
70  b5:4 PE2MD = 0 PE2 入出力 (ポート)
71  b3:2 PE1MD = 0 PE1 入出力 (ポート)
72  b1 Reserved
73  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
74  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
75
76  /* ATU-III(A)の設定 */
77  /* Configure TCRA
78  b7 EVOSEL2A = 0 イベント出力 2A セレクト
79  b6 EVOSEL2B = 0 イベント出力 2B セレクト
80  b5:3 EVOSEL1 = 001 タイマ B へ TIA00 の入力エッジを出力
81  b2:0 CKSELA = 000 カウントクロックにクロックバス 0(プリスケアラ 0)を選択 */
82  ATUA.TCRA.BYTE = 0x08; /* タイマ A をクロックバス 0 のカウントクロックで動作
83     TIA00 の入力をイベントとしてタイマ B に出力 */
84  /* Configure TIOR1A
85  b15:12 Reserved
86  b11:10 IOA5 = 00 TIA のインプットキャプチャ禁止
87  b9:8 IOA4 = 00 TIA のインプットキャプチャ禁止
88  b7:6 IOA3 = 00 TIA のインプットキャプチャ禁止
89  b5:4 IOA2 = 00 TIA のインプットキャプチャ禁止
90  b3:2 IOA1 = 00 TIA のインプットキャプチャ禁止
91  b1:0 IOA0 = 01 TIA の立ち上がりで ICRA にキャプチャ */
92  ATUA.TIOR1A.WORD = 0x0001; /* TIA00 入力の立ち上がりエッジで ICRA にインプットキャプチャ
93  */
94  /* ATU-III(B)の設定 */

```

```

95  /* Configure TIORB
96  b7 LDSEL = 0 TCNTB2 にロードする値に ICRB0 を用いる
97  b6 CTCNTB5 = 0 TCNTB5 のカウントを許可
98  b5 EVCNTB = 1 外部イベントの入力を許可
99  b4 LDEN = 0 リロードレジスタの更新を許可
100 b3 CCS = 0 TCNTB=TCNTB4 のとき、TCNTB4 のカウント動作を停止しない
101 b2:1 Reserved
102 b0 IOB6 = 1 TCNTB6 と OCB6 とのコンペアマッチを許可*/
103 ATUB.TIORB.BYTE = 0x21; /* TCNTB2 にロードする値を ICRB0 に設定。TCNTB5 のカウントを許
    可
104     外部イベントの入力を許可、リロードレジスタの更新を許可
105     TCNTB6 と OCB6 とのコンペアマッチを許可 */
106 /* Configure PIMR
107 b15:12 Reserved
108 b11:0 PIM = H'0004 クロックの通倍比を 4 倍に設定 */
109 ATUB.PIMR.WORD = 0x0004; /* 倍周クロックの通倍比を 4 倍に設定 */
110
111 /* Configure TCRB
112 b7:2 Reserved
113 b1:0 CKSELB = 00 カウントソースにクロックバス 0 を選択 */
114 ATUB.TCRB.BYTE = 0x00; /* TCNTB0, TCNTB2 をクロックバス 0 のカウントクロックで動作 */
115
116 /* Configure TIERB
117 b7:6 Reserved
118 b5:4 IREG = 01 CMFB6 有効後、次の外部イベント入力タイミングで割り込み要求を出力
119 b3 CMEB6 = 1 コンペアマッチ割り込み B6 要求の出力を許可
120 b2 CMEB1 = 0 コンペアマッチ割り込み B1 要求の出力を禁止
121 b1 ICEB0 = 0 インプットキャプチャ割り込み B0 要求の出力を禁止
122 b0 CMEB0 = 0 コンペアマッチ割り込み B0 要求の出力を禁止 */
123 ATUB.TIERB.BYTE = 0x18; /* TCNTB6, OCB6 のコンペアマッチ発生のタイミングでの割り込み要
    求を許可 */
124
125 /* Configure TCNTB
126 b31:0 入力エッジ間計測カウント値 = 0 */
127 ATUB.TCNTB6.LONG = 0x00000000; /* TCNTB6 の初期値を 0 に設定 */
128
129 /* Configure OCB6
130 b31:12 OCB6 = H'0002 アウトプットコンペア値を、2 に設定
131 b11:0 Reserved */
132 ATUB.OCRB6.LONG = 0x00002000; /* TCNTB6 に対応するコンペアマッチの値を 2 に設定 */
133
134 /* Configure TCNTB
135 b31:0 入力エッジ間計測カウント値 = 0 */
136 ATUB.TCNTB5.LONG = 0x00000000; /* TCNTB5 の初期値を 0 に設定 */
137
138 /* Configure TCCLRB
139 b31:12 CCLRB = 96 補正カウンタクリア値 B = 150
140 b11:0 Reserved */
141 ATUB.TCCLRB.LONG = 0x00096000; /* TCNTB5 の上限を 150 に設定 */
142

```

```

143  /* ATU-III(共通)の設定 */
144  /* Configure CBCNT
145  b7:6 Reserved
146  b5:4 CB4EG = 00 クロックバス 4:外部クロックのエッジ検出をしない
147  b3 Reserved
148  b2 CB5SEL = 1 クロックバス 5:タイマ B 出力の倍周補正クロック
149  b1:0 CB5EG = 00 クロックバス 5:外部クロックのエッジ検出をしない */
150  ATUCTRL.CBCNT.BYTE = 0x04; /* クロックバス 5 をタイマ B からの倍周補正クロック()に設定 */
151
152  /* Configure PSCR0
153  b15:10 Reserved
154  b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1) に設定 */
155  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラ 0 の分周比を 10 に設定 */
156
157  /* Configure IPR07
158  b15:12 ATU-B(CMIB0,CMIB1) = H'0 対応する割り込みの優先順位を 0 に設定
159  b11:8 ATU-B(CMIB6,ICIB0) = H'F 対応する割り込みの優先順位を 15 に設定
160  b7:4 ATU-C0(IMIC0-IMIC3) = H'0 対応する割り込みの優先順位を 0 に設定
161  b3:0 ATU-C0(OVIC0) = H'0 対応する割り込みの優先順位を 0 に設定 */
162  INTC.IPR07.WORD = 0x0F00; /* TCNTB6 のコンペアマッチの割り込み優先順位を 15 に設定 */
163
164  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
165
166  /* Configure ATUENR
167  b15:10 Reserved
168  b9 TJE = 0 タイマ J のカウント動作を停止
169  b8 THE = 0 タイマ H のカウント動作を停止
170  b7 TGE = 0 タイマ G のカウント動作を停止
171  b6 TFE = 0 タイマ F のカウント動作を停止
172  b5 TEE = 0 タイマ E のカウント動作を停止
173  b4 TDE = 0 タイマ D のカウント動作を停止
174  b3 TCE = 0 タイマ C のカウント動作を停止
175  b2 TBE = 1 タイマ B のカウント動作を許可
176  b1 TAE = 1 タイマ A のカウント動作を許可
177  b0 PSCE = 1 プリスケーラのカウント動作を許可 */
178  ATUCTRL.ATUENR.WORD = 0x0007; /* タイマ A,B およびプリスケーラのカウントをスタート */
179
180  while(1);
181  } /* End of function main() */
182
183  /*****
184  * Function Name : CMIB6
185  * Description   : コンペアマッチ割り込み
186  * Arguments     : none
187  * Return Value  : none
188  *****/
189  void CMIB6(void)
190  {
191  /* Configure TSRB
192  b7:4 Reserved

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```
193  b3 CMFB6 = 0 コンペアマッチフラグをクリア
194  b2 CMFB1 = 0 コンペアマッチフラグをクリア
195  b1 ICFB0 = 0 コンペアマッチフラグをクリア
196  b0 CMFB0 = 0 コンペアマッチフラグをクリア */
197  ATUB.TSRB.BYTE &= 0; /* コンペアマッチフラグをクリア */
198
199  /* Configure TCNTB
200  b31:0 入力エッジ間計測カウント値 = 0 */
201  ATUB.TCNTB3.LONG = 0x00000000; /* TCNTB3 をクリア */
202  } /* End of function CMIB6() */
```

2.27.6 SH7253 グループでご使用の場合の変更点

本文 2.27 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.27.5 サンプルプログラムリストの 74 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.28 動作例 27 倍周補正クロック信号の生成[タイマ B]

2.28.1 概要

TCNTB3、TCNTB4、TCNTB5、TCCLRB の各レジスタを使用する事で、一定の外部入力区間で生成するクロックの平均を、入力信号の変化に関わらず、PIMR の値に補正したクロックを生成します。生成されたパルスを倍周補正クロックと言い、他のタイマでも使用できるよう、クロックバス 5 のクロックソースとして設定します。

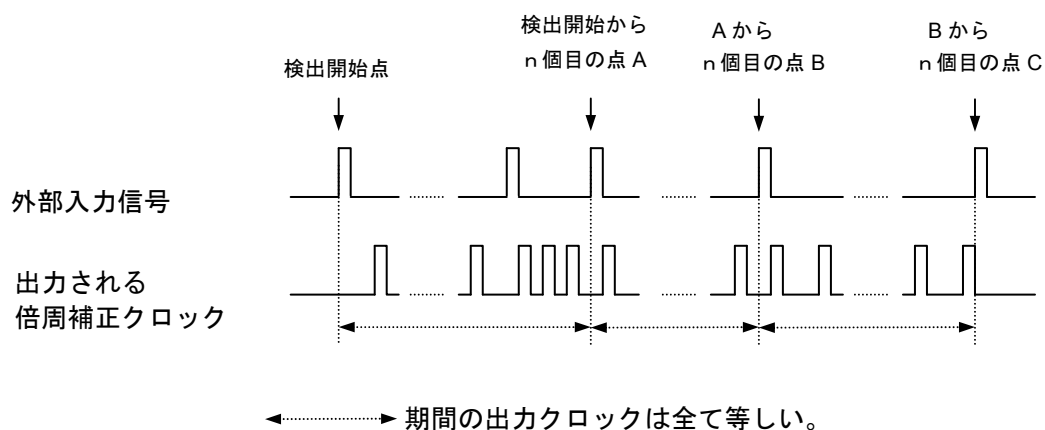


図 2.45 倍周補正クロック信号の生成(概要)

2.28.2 使用機能説明

表 2.29に使用端子及び関連レジスタの機能割り付けを示します。

表 2.29 ATU-III機能割り付け

使用端子		機能
端子	TIA00	倍周クロックの生成に使用する外部入力信号をこの端子に入力します。
	TOD00~02B	ワンショットパルスをこの端子から出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	CBCNT	クロックバス 5 のクロックソースを設定します。
タイマ A レジスタ	TCRA	タイマ A の動作クロック、外部入力エッジのイベント出力を設定します。
	TIOR1A	TIA1 の検出エッジを設定します。
タイマ B レジスタ	TIERB	割り込み許可の設定をします。
	TCRB	TCNTB0、TCNTB2 のカウントソースを設定します。
	TIORB	カウント動作の許可等、倍周クロック生成に関する設定を行います。
	PIMR	生成する倍周クロックの通倍比を設定します。
	TCNTB3	外部入力信号毎に TCNTB4 に値を転送し、PIMR の値が加算されます。
	TCNTB4	外部入力信号毎に TCNTB3 の値を読み込み、倍周クロックでアップカウント動作を行います。
	TCNTB5	TCNTB4 の値より小さいときに、PΦクロックでアップカウント動作を行います。
	TCCLRB	TCNTB5 の上限値として機能し、このレジスタと TCNTB5 が一致すると、TCNTB5 はクリアされます。
	TCNTB6	入力エッジ間計測カウントです。
	OCRB6	TCNTB6 に対応するコンペアマッチの値を設定します。
	TSRB	各コンペアマッチの発生状況を示します。
PFC レジスタ	PECR1	PE1 を TIA01 に設定します。
	PF0CR1	PF0,PF1,PF2 を TOD 出力 00B,01B,02B に設定します。
INTC レジスタ	IPR06	割り込みの優先順位を設定します。
	IPR07	割り込みの優先順位を設定します。
タイマ D レジスタ	TCRD	タイマ D の各カウンタの動作クロックバスを設定します。
	TIOR1D0	各コンペアマッチの発生の有無及び、発生時の動作を設定します。
	TSRD0	各コンペアマッチの発生状況を示します。
	DCNTD00~02	ダウンカウンタの値を設定します。
	DSTRD0	ダウンカウンタの動作許可を設定します。

2.28.3 動作説明

(1) 動作原理

図 2.46に減速時、図 2.47に加速時の動作原理を示します。これに従って SH72546R のハードウェア処理によって入力信号に対する倍周クロックの生成を行います。

[共通処理]

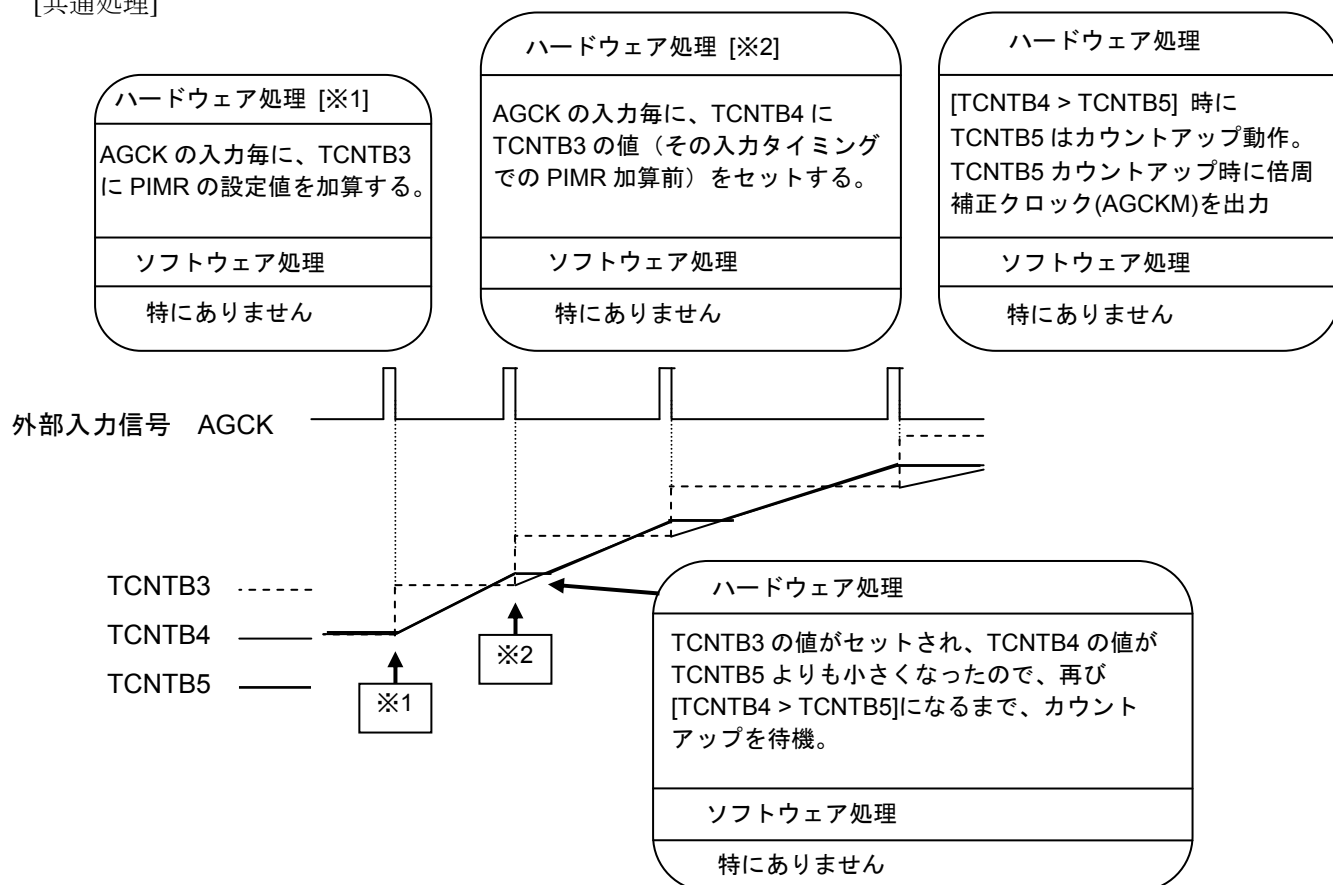


図 2.46 倍周補正クロック出力原理(減速時)

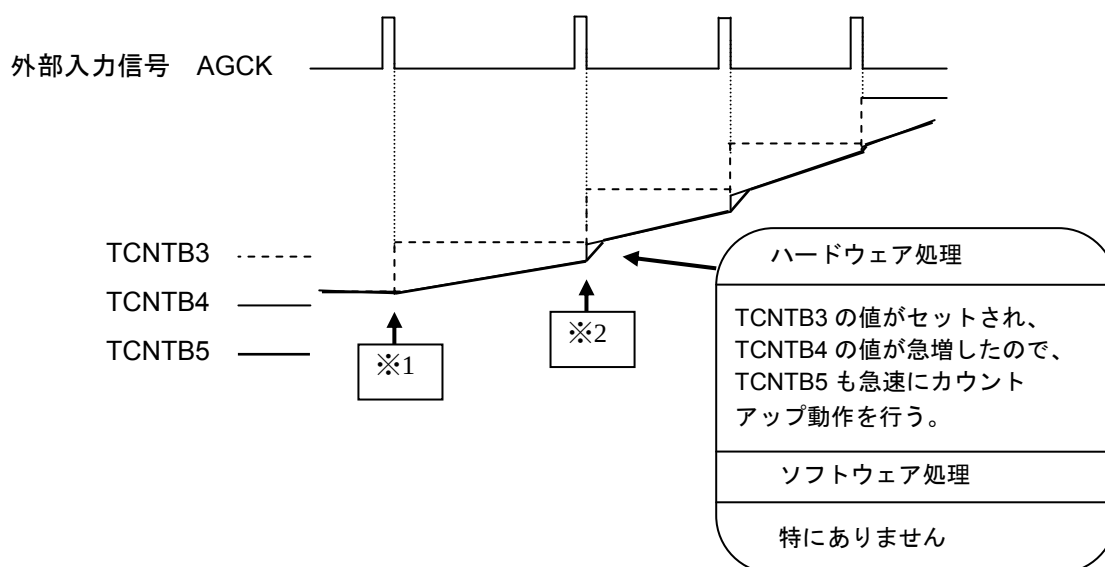


図 2.47 倍周補正クロック出力原理(加速時)

(2) 使用例

周期の変化する外部入力信号に対して、その信号から倍周クロックを生成する場合、補正を用いない場合と、用いる場合では下図のように動作が異なります。それぞれをクロックに用いて、例えば「ある外部入力から 12 カウント（外部入力 3 歯分）だけパルスを出力する」というソフトウェアを実行した場合の波形を図 2.48、図 2.49 に示します。（PIM の値が 4 の場合）

[1] 入力信号の間隔が増加していく場合（減速時）

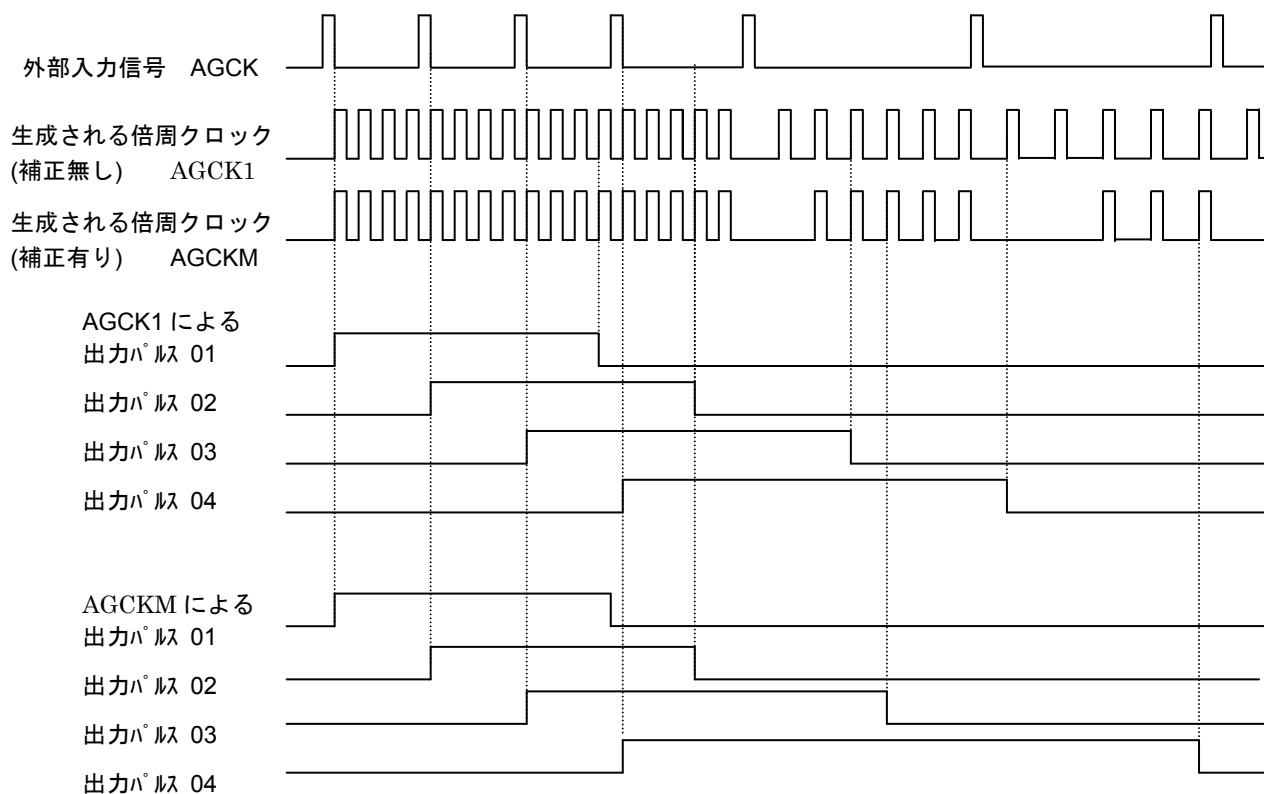


図 2.48 倍周機能使用に関する比較(減速時、PIM=4 の場合)

[2]入力信号の間隔が減少していく場合（加速時）

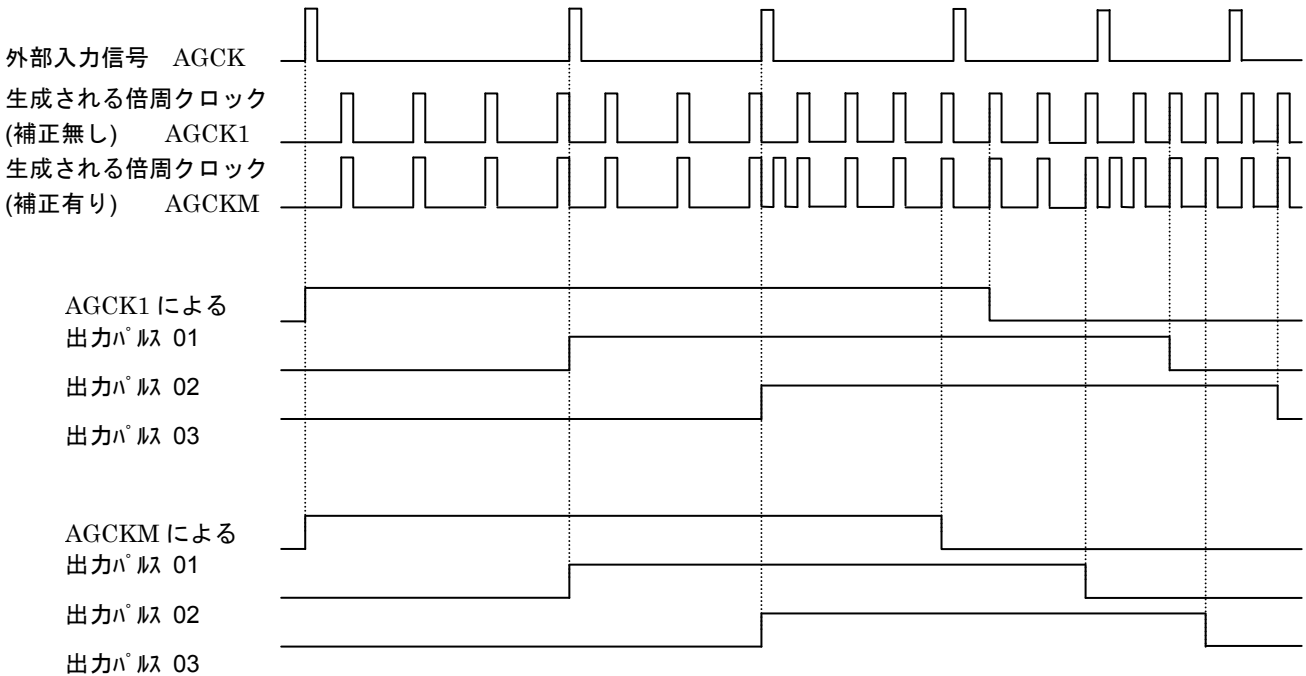


図 2.49 倍周機能使用に関する比較(加速時、PIM=4 の場合)

このように、倍周クロックを用いたカウント動作を基準にする事で、入力信号の波長が変化した場合に、波長の変化に対するソフト的な処理を行わなくても、波長に追従して処理を行う事ができます。そして、補正機能を使用する事により、より正確に外部入力信号の間隔に対応します。

2.28.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
インプットキャプチャ割り込み	CMIB6	TCTB6 と OCRB6 のコンペアマッチ割り込みが発生したら、コンペアマッチフラグをクリアします。
パルス出力ルーチン	CMID00	TCNTB1 と OCRB1 のコンペアマッチが発生したら、ワンショットパルスを出力します。

- 使用変数の説明

ラベル名	機能	データ長	使用モジュール
cnt	ワンショットパルス出力ルーチン参照の回数を記録します。	unsigned char	ワンショットパルス出力ルーチン

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E の PE0 端子を入力端子 TIA001 に設定します。	0x0001	メインルーチン
PFCR1	ポート F0,F1,F2 を TOD 出力 00B,01B,02B に設定します。	0x0015	
PSCR0	プリスケアラ 0 の分周比を 100 に設定します。	0x0063	
ATUENR	ATU-IIIチャネル A,B,及びプリスケアラのカウント動作を許可します。	0x0007	
CBCNT	クロックバス 5 のクロックソースをタイマ B の倍周補正クロックに設定します。	0x04	
TCRA	タイマ B へ TIA00 の入力エッジを出力します。	0x08	
TIERA	割り込みを許可します。	0x02	
TIOR1A	TIA00 への入力信号の立ち上がりエッジを検出する。	0x0001	
TIORB	TCNTB2 のリロード値に ICRB0 の値を使用、外部イベントの入力を許可、そして、TCNTB3=TCNTB4 の時に TCNTB4 のカウントを停止するよう設定します。	0x21	
PIMR	外部入力周期に対する倍周クロックの通倍比を 10 に設定します。	0x000A	
TCRB	TCNTB0,TCNTB2 の使用クロックをクロックバス 0 に設定します。	0x00	
TCCLRB	TCNTB5 の上限値を 104 に 設定します。	0x00068000	
TIERB	コンペアマッチ B6 の割り込み要求を許可します。	0x08	
TCNTB5	TCNTB5 の初期値を 0 に設定します。	0x00000000	
TCNTB6	TCNTB6 の初期値を 0 に設定します。	0x00000000	
OCRB6	TCNTB6 のコンペアマッチの値を 15 に設定します。	0x0000F000	コンペアマッチ割り込みルーチン ワンショットパルス出力ルーチン
IPR07	タイマ B に関する割り込みの優先度を設定します。	0x0F00	
TSRB	コンペアマッチフラグ CMFB6 をクリアします。	0x00	
TSRD0	コンペアマッチ、アンダーフローフラグをクリアします。	0x0000	
TCNTB3	補正イベントカウント B3 をクリアします。	0x00000000	
TCNT1D0	タイマカウンタ 1D0 をクリアします。	0x00000000	
DCNTD00~02	ダウンカウンタ 00~02 に 250 をセットします。	0x00FA<<8	
DSTRD0	ダウンカウンタの停止/開始を設定します。	0x01 0x02 0x04	

2.28.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description     : This is the main tutorial code.
33 * Operation      : Multiple correction clock signal generation.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45
46 /*****
47 Private global variables and functions
48 *****/

```

```

49  unsigned char cnt;
50  void main(void); /* メインルーチン */
51
52  /*****
53  * Function Name : main
54  * Description   : The main loop
55  * Arguments     : none
56  * Return Value  : none
57  *****/
58
59  void main(void)
60  {
61  /* ポートの設定 */
62  /* Configure PECR1
63  b15:14 PE7MD = 0 PE7 入出力 (ポート)
64  b13:12 PE6MD = 0 PE6 入出力 (ポート)
65  b11 Reserved
66  b10 PE5MD = 0 PE5 入出力 (ポート)
67  b9 Reserved
68  b8 PE4MD = 0 PE4 入出力 (ポート)
69  b7 Reserved
70  b6 PE3MD = 0 PE3 入出力 (ポート)
71  b5:4 PE2MD = 0 PE2 入出力 (ポート)
72  b3:2 PE1MD = 0 PE1 入出力 (ポート)
73  b1 Reserved
74  b0 PE0MD = 1 TIA00 入力 (ATU-III) */
75  PORTE.CR1.WORD = 0x0001; /* PE0 を TIA00 入力に設定 */
76
77  /* Configure PFCR1
78  b15:14 PF7MD = 00 PF7 モード = PF7 入出力 (ポート)
79  b13:12 PF6MD = 00 PF6 モード = PF6 入出力 (ポート)
80  b11:10 PF5MD = 00 PF5 モード = PF5 入出力 (ポート)
81  b9:8 PF4MD = 00 PF4 モード = PF4 入出力 (ポート)
82  b7:6 PF3MD = 00 PF3 モード = PF3 入出力 (ポート)
83  b5:4 PF2MD = 01 PF2 モード = TOD22B 出力 (ATU-III)
84  b3:2 PF1MD = 01 PF1 モード = TOD21B 出力 (ATU-III)
85  b1:0 PF0MD = 01 PF0 モード = TOD20B 出力 (ATU-III) */
86  PORTF.CR1.WORD = 0x0015; /* PF0,1,2 を TOD00B,01B,02B 出力に設定 */
87
88  /* ATU-III(A)の設定 */
89  /* Configure TCRA
90  b7 EVOSEL2A = 0 イベント出力 2A セレクト
91  b6 EVOSEL2B = 0 イベント出力 2B セレクト
92  b5:3 EVOSEL1 = 001 タイマ B へ TIA00 の入力エッジを出力
93  b2:0 CKSELA = 000 カウントクロックにクロックバス 0(プリスケアラ 0)を選択 */
94  ATUA.TCRA.BYTE = 0x08; /* タイマ A をクロックバス 0 のカウントクロックで動作
95     TIA00 の入力をイベントとして、タイマ B に出力 */
96  /* Configure TIOR1A
97  b15:12 Reserved
98  b11:10 IOA5 = 00 TIA05 のインプットキャプチャ禁止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

99      b9:8 IOA4 = 00 TIA04 のインプットキャプチャ禁止
100     b7:6 IOA3 = 00 TIA03 のインプットキャプチャ禁止
101     b5:4 IOA2 = 00 TIA02 のインプットキャプチャ禁止
102     b3:2 IOA1 = 00 TIA01 のインプットキャプチャ禁止
103     b1:0 IOA0 = 01 TIA00 の立ち上がりで ICRA にキャプチャ */
104     ATUA.TIOR1A.WORD = 0x0001; /* TIA00 の立ち上がりエッジでインプットキャプチャ */
105
106     /* ATU-III(B)の設定 */
107     /* Configure TIORB
108     b7 LDSEL = 0 TCNTB2 にロードする値に ICRB0 を用いる
109     b6 CTCNTB5 = 0 TCNTB5 のカウントを許可
110     b5 EVCNTB = 1 外部イベントの入力を許可
111     b4 LDEN = 0 リロードレジスタの更新を許可
112     b3 CCS = 0 TCNTB=TCNTB4 のとき、TCNTB4 のカウント動作を停止しない
113     b2:1 Reserved
114     b0 IOB6 = 1 TCNTB6 と OCB6 とのコンペアマッチを許可 */
115     ATUB.TIORB.BYTE = 0x21; /* TCNTB2 にロードする値を ICRB0 に設定。TCNTB5 のカウントを許
可
116     外部イベントの入力を許可、リロードレジスタの更新を許可
117     TCNTB6 と OCB6 とのコンペアマッチを許可 */
118     /* Configure PIMR
119     b15:12 Reserved
120     b11:0 PIM = H'000A クロックの通倍比を 10 倍に設定 */
121     ATUB.PIMR.WORD = 0x000A; /* 倍周クロックの通倍比を 10 倍に設定 */
122
123     /* Configure TCRB
124     b7:2 Reserved
125     b1:0 CKSELB = 00 カウントソースにクロックバス 0 を選択 */
126     ATUB.TCRB.BYTE = 0x00; /* TCNTB0, TCNTB2 をクロックバス 0 のカウントクロックで動作 */
127
128     /* Configure TIERB
129     b7:6 Reserved
130     b5:4 IREG = 00 CMFB6 が有効になったタイミングで割り込み要求出力
131     b3 CMEB6 = 1 コンペアマッチ割り込み B6 要求の出力を許可
132     b2 CMEB1 = 0 コンペアマッチ割り込み B1 要求の出力を禁止
133     b1 ICEB0 = 0 インプットキャプチャ割り込み B0 要求の出力を禁止
134     b0 CMEB0 = 0 コンペアマッチ割り込み B0 要求の出力を禁止 */
135     ATUB.TIERB.BYTE = 0x08; /* TCNTB6, OCB6 のコンペアマッチ発生タイミングでの割り込み要求を許可 */
136
137     /* Configure TCNTB
138     b31:0 入力エッジ間計測カウント値 = 0 */
139     ATUB.TCNTB6.LONG = 0x00000000; /* TCNTB6 の初期値を 0 に設定 */
140
141     /* Configure OCB6
142     b31:12 OCB6 = H'0000F アウトプットコンペア値を、15 に設定
143     b11:0 Reserved */
144     ATUB.OCRB6.LONG = 0x0000F000; /* TCNTB6 に対応するコンペアマッチの値を 15 に設定 */
145
146     /* Configure TCNTB
147     b31:0 入力エッジ間計測カウント値 = 0 */

```



```

148  ATUB.TCNTB5.LONG = 0x00000000; /* TCNTB5 の初期値を 0 に設定 */
149
150  /* Configure TCCLRB
151  b31:12 CCLRB = H'68 補正カウンタクリア値 B = D'104
152  b11:0 Reserved */
153  ATUB.TCCLRB.LONG = 0x00068000; /* TCNTB5 の上限を 104 に設定 */
154
155  /* ATU-III (共通) の設定 */
156  /* Configure CBCNT
157  b7:6 Reserved
158  b5:4 CB4EG = 00 クロックバス 4: 外部クロックのエッジ検出をしない
159  b3 Reserved
160  b2 CB5SEL = 1 クロックバス 5: タイマ B 出力の倍周補正クロック
161  b1:0 CB5EG = 00 クロックバス 5: 外部クロックのエッジ検出をしない */
162  ATUCTRL.CBCNT.BYTE = 0x04; /* クロックバス 5 をタイマ B からの倍周補正クロック() に設定 */
163
164  /* Configure PSCR0
165  b15:10 Reserved
166  b9:0 PSCn = H'63 プリスケーラの分周比を 100 (設定値は -1) に設定 */
167  ATUCTRL.PSCR0.WORD = 0x0063; /* プリスケーラ 0 の分周比を 100 に設定 */
168
169  /* Configure IPR07
170  b15:12 ATU-B(CMIB0, CMIB1) = H'0 対応する割り込みの優先順位を 0 に設定
171  b11:8 ATU-B(CMIB6, ICIB0) = H'F 対応する割り込みの優先順位を 15 に設定
172  b7:4 ATU-C0(IMIC0-IMIC3) = H'0 対応する割り込みの優先順位を 0 に設定
173  b3:0 ATU-C0(OVIC0) = H'0 対応する割り込みの優先順位を 0 に設定 */
174  INTC.IPR07.WORD = 0x0F00; /* TCNTB6 のコンペアマッチの割り込み優先順位を 15 に設定 */
175
176  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
177
178  /* Configure ATUENR
179  b15:10 Reserved
180  b9 TJE = 0 タイマ J のカウント動作を停止
181  b8 THE = 0 タイマ H のカウント動作を停止
182  b7 TGE = 0 タイマ G のカウント動作を停止
183  b6 TFE = 0 タイマ F のカウント動作を停止
184  b5 TEE = 0 タイマ E のカウント動作を停止
185  b4 TDE = 0 タイマ D のカウント動作を停止
186  b3 TCE = 0 タイマ C のカウント動作を停止
187  b2 TBE = 1 タイマ B のカウント動作を停止
188  b1 TAE = 1 タイマ A のカウント動作を許可
189  b0 PSCE = 1 プリスケーラのカウント動作を許可 */
190  ATUCTRL.ATUENR.WORD = 0x0007; /* タイマ A, B およびプリスケーラのカウントをスタート */
191
192  while(1);
193  } /* End of function main() */
194
195  /*****
196  * Function Name : CMIB6
197  * Description   : コンペアマッチ割り込み(B6)

```

```

198  * Arguments      : none
199  * Return Value   : none
200  *****/
201  void CMIB6(void)
202  {
203  /* Configure TSRB
204  b7:4 Reserved
205  b3 CMFB6 = 0 コンペアマッチフラグをクリア
206  b2 CMFB1 = 0 コンペアマッチフラグをクリア
207  b1 ICFB0 = 0 コンペアマッチフラグをクリア
208  b0 CMFB0 = 0 コンペアマッチフラグをクリア */
209  ATUB.TSRB.BYTE &= 0; /* コンペアマッチフラグをクリア */
210  } /* End of function CMIB6() */
211
212  /*****/
213  * Function Name : CMID00
214  * Description   : ワンショットパルス出力ルーチン
215  * Arguments     : none
216  * Return Value  : none
217  *****/
218  void CMID00(void)
219  {
220  /* Configure TSRD
221  b15:14 Reserved
222  b13 OVFDn = 0 TCNT2Dn フラグをオーバフローなしにクリア
223  b12 OVFDn = 0 TCNT1Dn フラグをオーバフローなしにクリア
224  b11 UDFDn3 = 0 UDFDn3 フラグをアンダーフローなしにクリア
225  b10 UDFDn2 = 0 UDFDn2 フラグをアンダーフローなしにクリア
226  b9 UDFDn1 = 0 UDFDn1 フラグをアンダーフローなしにクリア
227  b8 UDFDn0 = 0 UDFDn0 フラグをアンダーフローなしにクリア
228  b7 CMFADn3 = 0 CMFADn3 フラグをコンペアマッチなしにクリア
229  b6 CMFADn2 = 0 CMFADn2 フラグをコンペアマッチなしにクリア
230  b5 CMFADn1 = 0 CMFADn1 フラグをコンペアマッチなしにクリア
231  b4 CMFADn0 = 0 CMFADn0 フラグをコンペアマッチなしにクリア
232  b3 CMFBDn3 = 0 CMFBDn3 フラグをコンペアマッチなしにクリア
233  b2 CMFBDn2 = 0 CMFBDn3 フラグをコンペアマッチなしにクリア
234  b1 CMFBDn1 = 0 CMFBDn3 フラグをコンペアマッチなしにクリア
235  b0 CMFBDn0 = 0 CMFBDn3 フラグをコンペアマッチなしにクリア */
236  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンダフローフラグをクリア */
237
238  /* Configure TCNTB
239  b31:0 入力エッジ間計測カウンタ値 = 0 */
240  ATUB.TCNTB3.LONG = 0x00000000;
241
242  /* Configure TCNA1D
243  b31:8 タイマカウンタ値 = 0
244  b7:0 Reserved */
245  ATUD.SUBBLOCKA[0].TCNT1D = 0x00000000;
246
247  cnt ++;

```

```
248  if(cnt == 1)
249  {
250  /* Configure DCNTD00
251  b31:8 タイマダウンカウンタ値
252  b7:0 Reserved */
253  ATUD.SUBBLOCKB[0].DCNTD[0] = 0x00FA << 8; /* ダウンカウンタ 00 を 250 にセット */
254
255  /* Configure DSTRD0
256  b7:4 Reserved
257  b3 DSTD03=0 DCNTD03 ダウンカウント停止
258  b2 DSTD02=0 DCNTD02 ダウンカウント停止
259  b1 DSTD01=0 DCNTD01 ダウンカウント停止
260  b0 DSTD00=1 DCNTD00 ダウンカウント開始 */
261  ATUD.SUBBLOCKB[0].DSTRD.BYTE = 0x01; /* ダウンカウンタ 00 のダウンカウントを開始 */
262  }
263  else if(cnt == 4)
264  {
265  /* Configure DCNTD01
266  b31:8 タイマダウンカウンタ値
267  b7:0 Reserved */
268  ATUD.SUBBLOCKB[0].DCNTD[1] = 0x00FA << 8; /* ダウンカウンタ 01 を 250 にセット */
269
270  /* Configure DSTRD0
271  b7:4 Reserved
272  b3 DSTD03=0 DCNTD03 ダウンカウント停止
273  b2 DSTD02=0 DCNTD02 ダウンカウント停止
274  b1 DSTD01=1 DCNTD01 ダウンカウント開始
275  b0 DSTD00=0 DCNTD00 ダウンカウント停止 */
276  ATUD.SUBBLOCKB[0].DSTRD.BYTE = 0x02; /* ダウンカウンタ 01 のダウンカウントを開始 */
277  }
278  else if(cnt == 8)
279  {
280  /* Configure DCNTD02
281  b31:8 タイマダウンカウンタ値
282  b7:0 Reserved */
283  ATUD.SUBBLOCKB[0].DCNTD[2] = 0x00FA << 8; /* ダウンカウンタ 02 を 250 にセット */
284
285  /* Configure DSTRD0
286  b7:4 Reserved
287  b3 DSTD03=0 DCNTD03 ダウンカウント停止
288  b2 DSTD02=1 DCNTD02 ダウンカウント開始
289  b1 DSTD01=0 DCNTD01 ダウンカウント停止
290  b0 DSTD00=0 DCNTD00 ダウンカウント停止 */
291  ATUD.SUBBLOCKB[0].DSTRD.BYTE = 0x04; /* ダウンカウンタ 02 のダウンカウントを開始 */
292  }
293  else if(cnt == 10)
294  {
295  cnt = 0;
296  }
297  } /* End of function CMID00() */
```

2.28.6 SH7253 グループでご使用の場合の変更点

本文 2.28 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA00 端子が、SH7253 グループでは、PG12 端子（SH7254R グループでは、PE0 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.28.5 サンプルプログラムリストの 75 行目を

PORTG.CR4A.WORD=0x0001 に変更

SH7254R グループ : PE0CR を 0x0001 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0001 に設定 (変更後)

2.29 動作例 28 倍周補正クロック信号の生成[タイマ B]

2.29.1 概要

欠け歯の含まれた外部入力信号に対して、その外部入力信号で生成した倍周補正クロック信号でダウンカウント動作を行うワンショットパルスを出力する事で、欠け歯を基準として、一定の位置でパルスを出力するような出力信号を生成します。

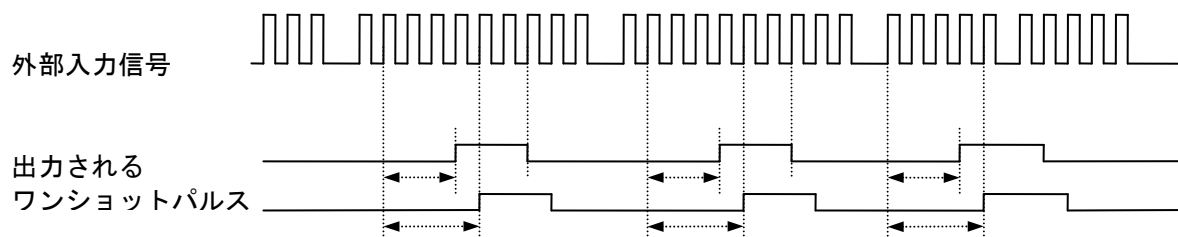


図 2.50 倍周補正クロック信号の生成(概要)

2.29.2 使用機能説明

表 2.30に使用端子及び関連レジスタの機能割り付けを示します。

表 2.30 ATU-III機能割り付け

使用端子		機能
端子	TIA01	倍周クロックの生成に使用する外部入力信号をこの端子に入力します。
	TOD00～02B	ワンショットパルスをこの端子から出力します。
関連レジスタ		機能
ATU-III 共通レジスタ	ATUENR	各タイマやプリスケアラの動作を設定します。
	PSCR0	プリスケアラ 0 の分周比を設定します。
	CBCNT	クロックバス 5 のクロックソースを設定します。
タイマ A レジスタ	TCRA	タイマ A の動作クロック、外部入力エッジのイベント出力を設定します。
	TIOR1A	TIA00 の検出エッジを設定します。
	TSRA	インプットキャプチャフラグが格納されます。
	TIERA	インプットキャプチャ割り込みの許可/禁止を設定します。
タイマ B レジスタ	TCNTB2	外部イベント入力により値が更新されます。
	LDB	TCNTB2、RLDB の更新に用いられます。
	RLDB	ICRB0 または、LDB の値を更新します。
	TIERB	割り込みの許可をします。
	TCNTB6	AGCKM によりカウントアップします。
	OCRB6	TCNTB6 と比較され一致したら、割り込みを発生します。
	TCRB	TCNTB0、TCNTB2 のカウントソースを設定します。
	TIORB	カウント動作の許可等、倍周クロック生成に関する設定を行います。
	PIMR	生成する倍周クロックの通倍比を設定します。
	TCNTB3	外部入力信号毎に TCNTB4 に値を転送し、PIMR の値が加算されます。
	TCNTB4	外部入力信号毎に TCNTB3 の値を読み込み、倍周クロックでアップカウント動作を行います。
	TCNTB5	TCNTB4 の値より小さいときに、PΦクロックでアップカウント動作を行います。
	TCCLRB	TCNTB5 の上限値として機能し、このレジスタと TCNTB5 が一致すると、TCNTB5 はクリアされます。
	TCNTB6	OCRB6 と一致することで、コンペアマッチ割り込みを発生します。
	OCRB6	TCNTB6 に対応するコンペアマッチの値を設定します。
	TSRB	タイマ B 各コンペアマッチ発生状況を示します。
INTC レジスタ	IPR10	割り込みの優先順位を設定します。
	IPR06	割り込みの優先順位を設定します。
	IPR07	割り込みの優先順位を設定します。
タイマ D レジスタ	TCRD0	タイマ D 各カウンタの動作クロックバスを設定します。
	TIOR1D0	各コンペアマッチの発生の有無及び、発生時の動作を設定します。
	DCRD0	ダウンカウンタの動作開始、停止の条件を設定します。
	TSRD0	各コンペアマッチの発生状況を示します。
	OCRD00～02	OCRD00 と TCNT1D0、OCRD01 と TCNT1D1、OCRD02 と TCNT1D2 に対応するコンペアマッチ A の条件を設定。
	TIERD0	OCRD00～02 とのコンペアマッチ発生による割り込み要求を設定します。
	TSTRD	タイマ D のカウントをスタートします。

2.29.3 動作説明

(1) 動作原理

図 2.51に減速時の、図 2.52に加速時の動作原理を示します。これに従って SH72546R のハードウェア処理によって入力信号に対する倍周クロックの生成を行います。

[共通処理]

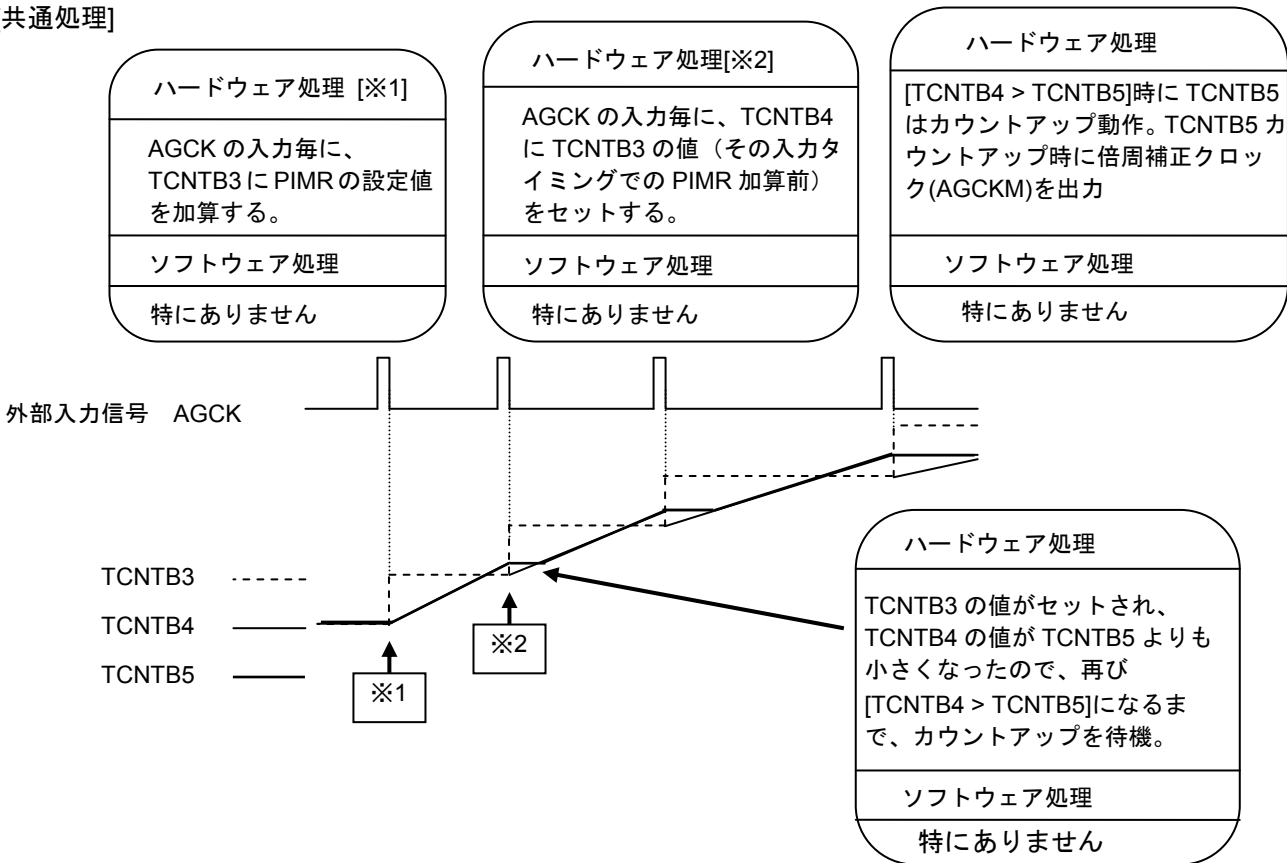


図 2.51 倍周補正クロック出力原理(減速時)

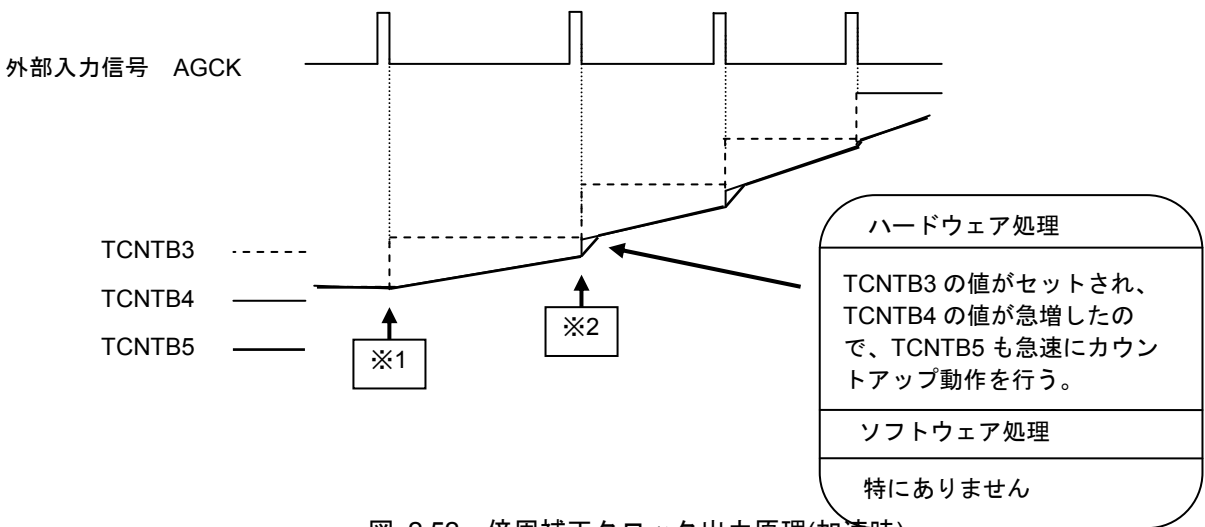


図 2.52 倍周補正クロック出力原理(加速時)

(2) 使用例

周期の変化する外部入力信号に対して、その信号から倍周クロックを生成する場合、補正を用いない場合と、用いる場合では下図のように動作が異なります。それぞれをクロックに用いて、例えば「ある外部入力から 12 カウント（外部入力 3 歯分）だけパルスを出力する」というソフトウェアを実行した場合の波形を図 2.53、図 2.54 に示します。（PIM の値が 4 の場合）

[1] 入力信号の間隔が増加していく場合（減速時）

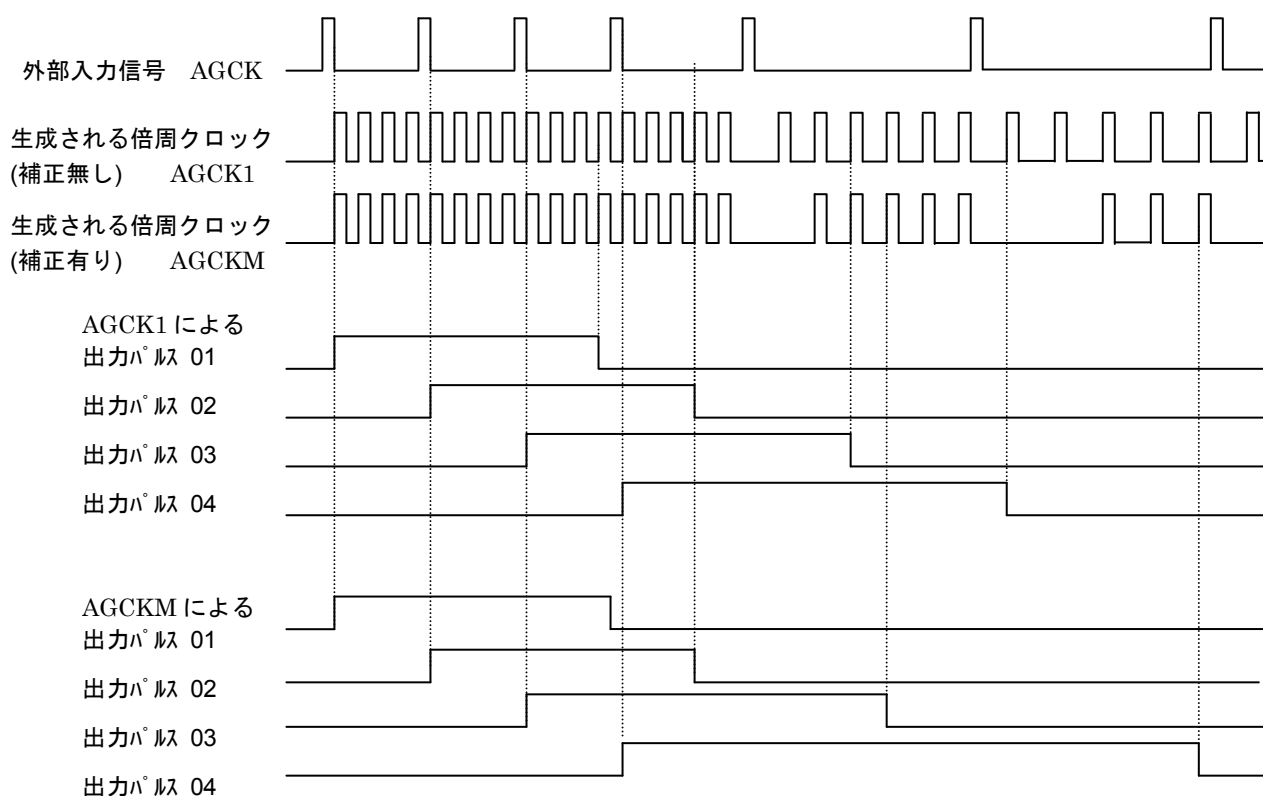


図 2.53 倍周機能使用に関する比較(減速時、PIM=4 の場合)

[2]入力信号の間隔が減少していく 場合（加速時）

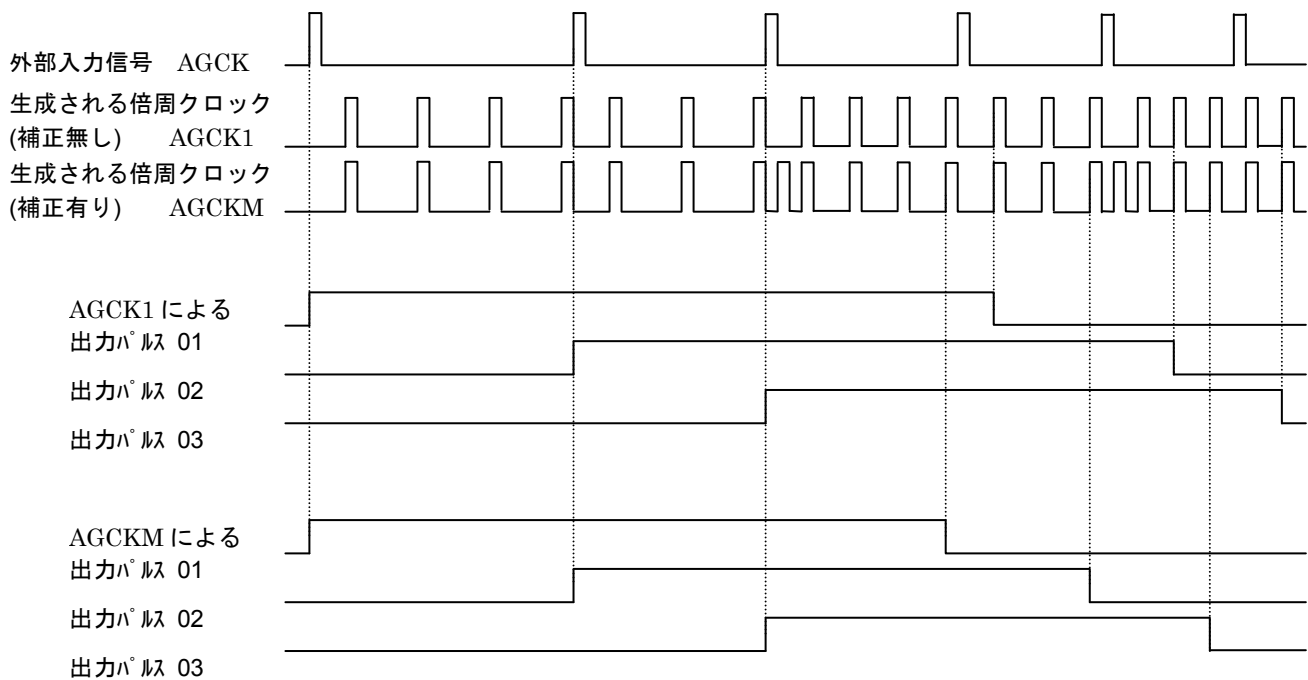


図 2.54 倍周機能使用に関する比較(加速時、PIM=4 の場合)

このように、倍周クロックを用いたカウント動作を基準にする事で、入力信号の波長が変化した場合に、波長の変化に対するソフト的な処理を行わなくても、波長に追従して処理を行う事ができます。そして、補正機能を使用する事により、より正確に外部入力信号の間隔に対応します。

2.29.4 ソフトウェア説明

- モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ATU-IIIの初期設定を行います。
外部入力検知ルーチン	ICIA1	外部入力検知したら本ルーチンを行います。
フラグクリアルーチン	CMID00	TCNTD1 と OCRD のコンペアマッチが発生したら、本ルーチンを行います。
パルス出力ルーチン	CMIB6	TCNTB6 と OCRB6 のコンペアマッチが発生したら、ワンショットパルスを出力します。

- 使用変数の説明

本タスクでは変数は使用していません。

- 使用レジスタの説明

レジスタ名	機能	設定値	使用モジュール名
PECR1	ポート E PE1 端子を入力端子 TIA1 に設定します。	0x0004	メインルーチン
PFCR1	ポート F0,F1,F2 を TOD 出力 00B,01B,02B に設定します。	0x0015	
PSCR0	プリスケアラ 0 の分周比 10 に設定します。	0x0009	
ATUENR	ATU-III チャンネル A,B,D 及びプリスケアラのカウント動作を許可します。	0x0017	
CBCNT	クロックバス 5 のクロックソースをタイマ B の倍周補正クロックに設定します。	0x04	
TCRA	TIA1 への入力信号をイベントとしてタイマ B に出力します。	0x10	
TIOR1A	TIA1 への入力信号の立ち上がりエッジを検出する。	0x0004	
TSRA	インプットキャプチャフラグをクリアします。	0x00	
TIERA	TIA00 のインプットキャプチャ割り込みを許可します。	0x01	
TIORB	TCNTB2 のリロード値に ICRB0 の値を使用、外部イベントの入力を許可、そして、TCNTB3=TCNTB4 の時に TCNTB4 のカウントを停止しないよう設定します。	0x21	
PIMR	外部入力周期に対する倍周クロックの通倍比を 4 に設定します。	0x0004	
OCRB6	TCNTB6 に対応するコンペアマッチの値を 6 に設定	0x00006000	
LDB	LDB の値を 0x7D0 に設定します。	0x000007D0	
TCRB	TCNTB0,TCNTB2 の使用クロックをクロックバス 0 に設定 します。	0x00	
TCCLRB	TCNTB5 の上限値を 0x28(= PIMR × 設定する外部入力数)に 設定します。	0x00028000	
TIERB	コンペアマッチ B6 の割り込み要求を許可します。	0x08	
TCNTB5	倍周補正クロック生成カウンタ B5 を 0 に設定します。	0x00000000	
TCNTB6	倍周クロックカウンタ B6 の初期値を 0 に設定します。	0x00000000	
TCRD0	TCNT1Dn、DCNTD0n が、それぞれクロックバス 5 でアップ/ダウンカウントを行うよう設定します。	0x1555	
DCRD0	コンペアマッチ A 発生時、ダウンカウント開始します。	0x0222	
TSRD0	コンペアマッチフラグをクリアします。	0x0000	
OCRD00 ~02	オフセット値を設定します。	-	
TIERD	割り込みの許可をします。	0x0010	
TSTRD	タイマ D のサブブロック 0 のカウントをスタートします。	0x01	
TIOR1D0	OCRD00、01 のコンペアマッチを許可します。	0x153F	
IPR06	タイマ A に関する割り込みの優先度を設定します。	0xF000	

SH7253 SH7254R グループ

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

IPR10	タイマ D に関する割り込みの優先度を設定します。	0xF000	パルス出力ルーチン
IPR07	タイマ B に関する割り込みの優先度を設定します。	0x0F00	
TSRB	タイマ B のコンペアマッチフラグをクリアします。	0x00	
TSRA	タイマ A のコンペアマッチフラグをクリアします。	0x00	
TIERA	TIA01 のインプットキャプチャ割り込みを許可します。	0x02	
TCNTB3	補正イベントカウンタ値を 0 に設定します。	0x00000000	
TIORB	TCNTB2 のロードデータ、RLDB のロードデータの計算に LDB 値を用いるよう設定します。	LDSEL=0x1	
DCNTD00~02	ダウンカウンタをセットします。	0x0800	フラグクリアルーチン
TSRD0	コンペアマッチ、アンダーフローフラグをクリアします。	0x0000	
TSRA	インプットキャプチャフラグをクリアします。	0x00	外部入力検知ルーチン
TIERA	TIA のインプットキャプチャ割り込みを禁止します。	0x00	
TIORB	TCNTB2 のロードデータ、RLDB のロードデータの計算に ICRB0 値を用いるよう設定します。	LDSEL=0x0	

2.29.5 サンプルプログラム

```

1  /*****
2  * DISCLAIMER
3  * This software is supplied by Renesas Electronics Corporation and is only
4  * intended for use with Renesas products. No other uses are authorized. This
5  * software is owned by Renesas Electronics Corporation and is protected under
6  * all applicable laws, including copyright laws.
7  * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
8  * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
9  * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
10 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
11 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
12 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
13 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
14 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
15 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
16 * Renesas reserves the right, without notice, to make changes to this software
17 * and to discontinue the availability of this software. By using this software,
18 * you agree to the additional terms and conditions found by accessing the
19 * following link:
20 * http://www.renesas.com/disclaimer *
21 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
22 *****/
23 /*****
24 * File Name      : SH7254R_ATU.c
25 * Version        : 1.00
26 * Device(s)      : SH72546R
27 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
28 *                 : C/C++ compiler package for the SuperH RISC engine family
29 *                 : (Ver.9.04 Release00).
30 * OS             : None
31 * H/W Platform   : SH7254R
32 * Description    : This is the main tutorial code.
33 * Operation      : Multiple correction clock signal generation.
34 *****/
35 /*****
36 * History : DD.MM.YYYY Version Description
37 *         : 12.01.2012 1.00 First Release
38 *****/
39
40 /*****
41 Includes <System Includes> , "Project Includes"
42 *****/
43 #include <machine.h> /* ライブラリ関数用ヘッダファイル */
44 #include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
45

```

```

46  /*****
47  Macro definitions
48  *****/
49  #define ofset0  0x0800          /* オフセット */
50  #define ofset1  0x0F00          /* オフセット */
51  #define ofset2  0x1400          /* オフセット */
52  #define puls_w  0x0800          /* パルス幅 */
53
54  /*****
55  Private global variables and functions
56  *****/
57
58  void main(void); /* メインルーチン */
59
60  /*****
61  * Function Name : main
62  * Description   : The main loop
63  * Arguments     : none
64  * Return Value  : none
65  *****/
66  void main(void)
67  {
68  /* ポートの設定 */
69  /* Configure PECR1
70  b15:14 PE7MD = 0 PE7 入出力 (ポート)
71  b13:12 PE6MD = 0 PE6 入出力 (ポート)
72  b11 Reserved
73  b10 PE5MD = 0 PE5 入出力 (ポート)
74  b9 Reserved
75  b8 PE4MD = 0 PE4 入出力 (ポート)
76  b7 Reserved
77  b6 PE3MD = 0 PE3 入出力 (ポート)
78  b5:4 PE2MD = 0 PE2 入出力 (ポート)
79  b3:2 PE1MD = 01 TIA01 入力 (ATU-III)
80  b1 Reserved
81  b0 PE0MD = 0 PE0 入出力 (ポート) */
82  PORTE.CR1.WORD = 0x0004; /* PE1 を TIA01 入力に設定 */
83
84  /* Configure PFCR1
85  b15:14 PF7MD = 00 PF7 モード = PF7 入出力 (ポート)
86  b13:12 PF6MD = 00 PF6 モード = PF6 入出力 (ポート)
87  b11:10 PF5MD = 00 PF5 モード = PF5 入出力 (ポート)
88  b9:8 PF4MD = 00 PF4 モード = PF4 入出力 (ポート)
89  b7:6 PF3MD = 00 PF3 モード = PF3 入出力 (ポート)
90  b5:4 PF2MD = 01 PF2 モード = TOD22B 出力 (ATU-III)
91  b3:2 PF1MD = 01 PF1 モード = TOD21B 出力 (ATU-III)
92  b1:0 PF0MD = 01 PF0 モード = TOD20B 出力 (ATU-III) */
93  PORTF.CR1.WORD = 0x0015; /* PF0,1,2 を TOD00B,01B,02B 出力に設定 */
94

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

95  /* ATU-III(A)の設定 */
96  /* Configure TCRA
97  b7 EVOSEL2A = 0 イベント出力 2A セレクト
98  b6 EVOSEL2B = 0 イベント出力 2B セレクト
99  b5:3 EVOSEL1 = 010 TIA01 の入力エッジを出力
100 b2:0 CKSELA = 000 カウントクロックにクロックバス 0(プリスケアラ 0)を選択 */
101 ATUA.TCRA.BYTE = 0x10; /* タイマ A をクロックバス 0 のカウントクロックで動作
102     TIA01 の入力をイベントとして、タイマ B に出力 */
103 /* Configure TIOR1A
104 b15:12 Reserved
105 b11:10 IOA5 = 00 TIA05 のインプットキャプチャ禁止
106 b9:8 IOA4 = 00 TIA04 のインプットキャプチャ禁止
107 b7:6 IOA3 = 00 TIA03 のインプットキャプチャ禁止
108 b5:4 IOA2 = 00 TIA02 のインプットキャプチャ禁止
109 b3:2 IOA1 = 01 TIA01 TIA の立ち上がりで ICRA にキャプチャ
110 b1:0 IOA0 = 00 TIA00 のインプットキャプチャ禁止 */
111 ATUA.TIOR1A.WORD = 0x0004; /* TIA01 の立ち上がりエッジでインプットキャプチャ */
112
113 /* Configure TSRA
114 b7 OVFA = 0 インプットキャプチャフラグをクリア
115 b6 Reserved
116 b5 ICFA5 = 0 インプットキャプチャフラグをクリア
117 b4 ICFA4 = 0 インプットキャプチャフラグをクリア
118 b3 ICFA3 = 0 インプットキャプチャフラグをクリア
119 b2 ICFA2 = 0 インプットキャプチャフラグをクリア
120 b1 ICFA1 = 0 インプットキャプチャフラグをクリア
121 b0 ICFA0 = 0 インプットキャプチャフラグをクリア */
122 ATUA.TSRA.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
123
124 /* Configure TIERA
125 b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
126 b6 Reserved
127 b5 ICEA5 = 0 インプットキャプチャ割り込み A5 要求の出力を禁止
128 b4 ICEA4 = 0 インプットキャプチャ割り込み A4 要求の出力を禁止
129 b3 ICEA3 = 0 インプットキャプチャ割り込み A3 要求の出力を禁止
130 b2 ICEA2 = 0 インプットキャプチャ割り込み A2 要求の出力を禁止
131 b1 ICEA1 = 0 インプットキャプチャ割り込み A1 要求の出力を禁止
132 b0 ICEA0 = 1 インプットキャプチャ割り込み A0 要求の出力を許可 */
133 ATUA.TIERA.BYTE = 0x01; /* TIA00 のインプットキャプチャ割り込みを許可 */
134
135 /* Configure IPR06
136 b15:12 ATU-A(ICIA0,1) = H'F 対応する割り込みの優先順位を 15 に設定
137 b11:8 ATU-A(ICIA2,3) = H'0 対応する割り込みの優先順位を 0 に設定
138 b7:4 ATU-A(ICIA4,5) = H'0 対応する割り込みの優先順位を 0 に設定
139 b3:0 ATU-A(OVIA) = H'0 対応する割り込みの優先順位を 0 に設定 */
140 INTC.IPR06.WORD = 0xF000; /* TIA00 の割り込み(ICIA0)の優先順位を 15 に設定 */
141
142 /* ATU-III(B)の設定 */
143 /* Configure TIORB
144 b7 LDSEL = 0 TCNTB2 にロードする値に ICRB0 を用いる

```

アドバンスタイマユニット-III (ATU-III) を使用したタイマ動作例

```

145  b6 CTCNTB5 = 0 TCNTB5 のカウントを許可
146  b5 EVCNTB = 1 外部イベントの入力を許可
147  b4 LDEN = 0 リロードレジスタの更新を許可
148  b3 CCS = 0 TCNTB=TCNTB4 のとき、TCNTB4 のカウント動作を停止しない
149  b2:1 Reserved
150  b0 IOB6 = 1 TCNTB6 と OCRB6 とのコンペアマッチを許可 */
151  ATUB.TIORB.BYTE = 0x21; /* TCNTB2 にロードする値を ICRB0 に設定。TCNTB5 のカウントを許可
152      外部イベントの入力を許可、リロードレジスタの更新を許可
153      TCNTB6 と OCRB6 とのコンペアマッチを許可 */
154  /* Configure PIMR
155  b15:12 Reserved
156  b11:0 PIM = H'0004 クロックの通倍比を 4 倍に設定 */
157  ATUB.PIMR.WORD = 0x0004; /* 倍周クロックの通倍比を 4 倍に設定 */
158
159  /* Configure TCRB
160  b7:2 Reserved
161  b1:0 CKSELB = 00 カウントソースにクロックバス 0 を選択 */
162  ATUB.TCRB.BYTE = 0x00; /* TCNTB0, TCNTB2 をクロックバス 0 のカウントクロックで動作 */
163
164  /* Configure TIERB
165  b7:6 Reserved
166  b5:4 IREG = 00 CMFB6 が有効になったタイミングで割り込み要求出力
167  b3 CMEB6 = 1 コンペアマッチ割り込み B6 要求の出力を許可
168  b2 CMEB1 = 0 コンペアマッチ割り込み B1 要求の出力を禁止
169  b1 ICEB0 = 0 インพุットキャプチャ割り込み B0 要求の出力を禁止
170  b0 CMEB0 = 0 コンペアマッチ割り込み B0 要求の出力を禁止 */
171  ATUB.TIERB.BYTE = 0x08; /* TCNTB6, OCRB6 のコンペアマッチ発生時のタイミングでの割り込み要求を許可 */
172
173  /* Configure TCNTB
174  b31:0 入力エッジ間計測カウント値 = 0 */
175  ATUB.TCNTB6.LONG = 0x00000000; /* TCNTB6 の初期値を 0 に設定 */
176
177  /* Configure OCRB6
178  b31:12 OCB6 = H'00006 アウトプットコンペア値を、6 に設定
179  b11:0 Reserved */
180  ATUB.OCRB6.LONG = 0x00000600; /* TCNTB6 に対応するコンペアマッチの値を 6 に設定 */
181
182  /* Configure LDB
183  b31:24 Reserved
184  b23:0 LDVAL = 0007D0 タイマロード値 = 2000 */
185  ATUB.LDB.LONG=0x000007D0;
186
187  /* Configure TCNTB
188  b31:0 入力エッジ間計測カウント値 = 0 */
189  ATUB.TCNTB5.LONG = 0x00000000; /* TCNTB5 の初期値を 0 に設定 */
190
191  /* Configure TCCLRB
192  b31:12 CCLRB = H'28 補正カウンタクリア値 B = D'40
193  b11:0 Reserved */
194  ATUB.TCCLRB.LONG = 0x00028000; /* TCCLRB の初期値を 40 に設定 */

```

```

195
196 /* ATU-III(D)の設定 */
197 /* Configure TCRD
198 b15 Reserved
199 b14 OBREDn = 0 OSBRDn のインプットキャプチャ動作を禁止
200 b13 C2CEDn = 0 タイマ B からの TCNT2Dn カウンタ値クリア要求を禁止
201 b12 C1CEDn = 1 タイマ B からの TCNT1Dn カウンタ値クリア要求を許可
202 b11 Reserved
203 b10:8 CKSEL2Dn = 101 クロックバス 5 で TCNT2Dn をアップカウント
204 b7 Reserved
205 b6:4 CKSEL1Dn = 101 クロックバス 5 で TCNT1Dn をアップカウント
206 b3 Reserved
207 b2:0 DCSELNn = 101 クロックバス 5 で DCNTDnm をダウンカウント */
208 ATUD.SUBBLOCKA[0].TCRD.WORD = 0x1555; /* TCNT1Dn をクロックバス 5 でアップカウント
209     DCNTD0n をクロックバス 5 でダウンカウント */
210 /* Configure TIOR1D
211 b15:14 OSSDn3 = 00 TODn3A 端子出力なし
212 b13:12 OSSDn2 = 01 コンペアマッチ A 発生時 TODn2A 端子に I/O コントロールビット A での設定値
    を出力
213 b11:10 OSSDn1 = 01 コンペアマッチ A 発生時 TODn1A 端子に I/O コントロールビット A での設定値
    を出力
214 b9:8 OSSDn0 = 01 コンペアマッチ A 発生時 TODn0A 端子に I/O コントロールビット A での設定値を
    出力
215 b7:6 IOADn3 = 00 OCRDn3 コンペアマッチ禁止
216 b5:4 IOADn2 = 11 OCRDn2 コンペアマッチ時、トグル出力
217 b3:2 IOADn1 = 11 OCRDn1 コンペアマッチ時、トグル出力
218 b1:0 IOADn0 = 11 OCRDn0 コンペアマッチ時、トグル出力 */
219 ATUD.SUBBLOCKB[0].TIOR1D.WORD = 0x153F; /* OCRD00 のコンペアマッチ A 発生時、TOD00A 端
    子から 0 出力 */
220
221 /* Configure DCRD
222 b15 Reserved
223 b14:12 TRGSELDn3 = 000 D3 ダウンカウントスタート/ストップトリガ選択 : なし
224 b11 Reserved
225 b10:8 TRGSELDn2 = 010 D2 ダウンカウントスタート/ストップトリガ選択 : コンペアマッチ A
226 b7 Reserved
227 b6:4 TRGSELDn1 = 010 D1 ダウンカウントスタート/ストップトリガ選択 : コンペアマッチ A
228 b3 Reserved
229 b2:0 TRGSELDn0 = 010 D0 ダウンカウントスタート/ストップトリガ選択 : コンペアマッチ A */
230 ATUD.SUBBLOCKB[0].DCRD.WORD = 0x0222; /* コンペアマッチ A 発生時にダウンカウントスタート
    */
231
232 /* Configure TSRD
233 b15:14 Reserved
234 b13 OVFDn = 0 オーバフローフラグをクリア
235 b12 OVFDn = 0 オーバフローフラグをクリア
236 b11 UDFDn3 = 0 アンダーフローフラグをクリア
237 b10 UDFDn2 = 0 アンダーフローフラグをクリア
238 b9 UDFDn1 = 0 アンダーフローフラグをクリア
239 b8 UDFDn0 = 0 アンダーフローフラグをクリア
240 b7 CMFADn3 = 0 コンペアマッチフラグをクリア

```


アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

241  b6 CMFADn2 = 0 コンペアマッチフラグをクリア
242  b5 CMFADn1 = 0 コンペアマッチフラグをクリア
243  b4 CMFADn0 = 0 コンペアマッチフラグをクリア
244  b3 CMFBDn3 = 0 コンペアマッチフラグをクリア
245  b2 CMFBDn2 = 0 コンペアマッチフラグをクリア
246  b1 CMFBDn1 = 0 コンペアマッチフラグをクリア
247  b0 CMFBDn0 = 0 コンペアマッチフラグをクリア */
248  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンドフローフラグをクリア */
249
250  /* Configure OCRD
251  b31:8 アウトプットコンペアレジスタ設定
252  b7:0 Reserved */
253  ATUD.SUBBLOCKB[0].OCRD[0] = (offset0); /* OCRD00 に TCNT1D0 に対応するコンペアマッチ A
    の条件を設定 */
254  ATUD.SUBBLOCKB[0].OCRD[1] = (offset1); /* OCRD01 に TCNT1D1 に対応するコンペアマッチ A
    の条件を設定 */
255  ATUD.SUBBLOCKB[0].OCRD[2] = (offset2); /* OCRD02 に TCNT1D2 に対応するコンペアマッチ A
    の条件を設定 */
256
257  /* Configure TIERD
258  b15:14 Reserved
259  b13 OVE2Dn = 0 OVF2Dn による割り込み要求を禁止
260  b12 OVE1Dn = 0 OVF1Dn による割り込み要求を禁止
261  b11 UDEDn3 = 0 UDFDn3 による割り込み要求を禁止
262  b10 UDEDn2 = 0 UDFDn2 による割り込み要求を禁止
263  b9 UDEDn1 = 0 UDFDn1 による割り込み要求を禁止
264  b8 UDEDn0 = 0 UDFDn0 による割り込み要求を禁止
265  b7 CMEADn3 = 0 CMFADn3 による割り込み要求を禁止
266  b6 CMEADn2 = 0 CMFADn2 による割り込み要求を禁止
267  b5 CMEADn1 = 0 CMFADn1 による割り込み要求を禁止
268  b4 CMEADn0 = 1 CMFADn0 による割り込み要求を許可
269  b3 CMEBDn3 = 0 CMFBDn3 による割り込み要求を禁止
270  b2 CMEBDn2 = 0 CMFBDn2 による割り込み要求を禁止
271  b1 CMEBDn1 = 0 CMFBDn1 による割り込み要求を禁止
272  b0 CMEBDn0 = 0 CMFBDn0 による割り込み要求を禁止 */
273  ATUD.SUBBLOCKB[0].TIERD.WORD = 0x0010; /* OCRD00 のコンペアマッチによる割り込みを許可 */
274
275  /* Configure TSTRC
276  b7:5 Reserved
277  b4 STRD4 = 0 TCNTD4 のカウント動作を停止
278  b3 STRD3 = 0 TCNTD3 のカウント動作を停止
279  b2 STRD2 = 0 TCNTD2 のカウント動作を停止
280  b1 STRD1 = 0 TCNTD1 のカウント動作を停止
281  b0 STRD0 = 1 TCNTD0 のカウント動作を許可 */
282  ATUD.TSTRD.BYTE = 0x01; /* タイマ D のサブブロック 0 のカウントをスタート */
283
284  /* ATU-III (共通) の設定 */
285  /* Configure CBCNT
286  b7:6 Reserved
287  b5:4 CB4EG = 00 クロックバス 4:外部クロックのエッジ検出をしない

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```

288  b3 Reserved
289  b2 CB5SEL = 1 クロックバス 5:タイマ B 出力の倍周補正クロック
290  b1:0 CB5EG = 00 クロックバス 5:外部クロックのエッジ検出をしない */
291  ATUCTRL.CBCNT.BYTE = 0x04; /* クロックバス 5 をタイマ B からの倍周補正クロック ( ) に設定 */
292
293  /* Configure PSCR0
294  b15:10 Reserved
295  b9:0 PSCn = H'9 プリスケーラの分周比を 10(設定値は-1) に設定 */
296  ATUCTRL.PSCR0.WORD = 0x0009; /* プリスケーラ 0 の分周比を 10 に設定 */
297
298  /* Configure IPR07
299  b15:12 ATU-B(CMIB0,CMIB1) = H'0 対応する割り込みの優先順位を 0 に設定
300  b11:8 ATU-B(CMIB6,ICIB0) = H'F 対応する割り込みの優先順位を 15 に設定
301  b7:4 ATU-C0(IMIC0-IMIC3) = H'0 対応する割り込みの優先順位を 0 に設定
302  b3:0 ATU-C0(OVIC0) = H'0 対応する割り込みの優先順位を 0 に設定 */
303  INTC.IPR07.WORD = 0x0F00; /* TCNTB6 のコンペアマッチの割り込み優先順位を 15 に設定 */
304
305  /* Configure IPR10
306  b15:12 ATU-D0(CMID00-CMID03) = H'F 対応する割り込みの優先順位を 15 に設定
307  b11:8 ATU-D0(OVI1D0-OVI2D0) = H'0 対応する割り込みの優先順位を 0 に設定
308  b7:4 ATU-D0(UDID00-UDID03) = H'0 対応する割り込みの優先順位を 0 に設定
309  b3:0 ATU-D1(CMID10-CMID13) = H'0 対応する割り込みの優先順位を 0 に設定 */
310  INTC.IPR10.WORD = 0xF000; /* OCRD00 の割り込み優先順位を 15 に設定 */
311
312  set_imask(0); /* 割り込みマスクレベルを 0 に設定 */
313
314  /* Configure ATUENR
315  b15:10 Reserved
316  b9 TJE = 0 タイマ J のカウント動作を停止
317  b8 THE = 0 タイマ H のカウント動作を停止
318  b7 TGE = 0 タイマ G のカウント動作を停止
319  b6 TFE = 0 タイマ F のカウント動作を停止
320  b5 TEE = 0 タイマ E のカウント動作を停止
321  b4 TDE = 1 タイマ D のカウント動作を許可
322  b3 TCE = 0 タイマ C のカウント動作を停止
323  b2 TBE = 1 タイマ B のカウント動作を許可
324  b1 TAE = 1 タイマ A のカウント動作を許可
325  b0 PSCE = 1 プリスケーラのカウント動作を許可 */
326  ATUCTRL.ATUENR.WORD = 0x0017; /* タイマ A, B, D およびプリスケーラのカウントをスタート */
327
328  while(1);
329  } /* End of function main() */
330
331  /*****
332  * Function Name : CMIB6
333  * Description   : コンペアマッチ割り込み(B6)
334  * Arguments    : none
335  * Return Value  : none
336  *****/
337  void CMIB6(void)

```

```

338  {
339  /* Configure TSRB
340  b7:4 Reserved
341  b3 CMFB6 = 0 コンペアマッチフラグをクリア
342  b2 CMFB1 = 0 コンペアマッチフラグをクリア
343  b1 ICFB0 = 0 コンペアマッチフラグをクリア
344  b0 CMFB0 = 0 コンペアマッチフラグをクリア */
345  ATUB.TSRB.BYTE &= 0; /* コンペアマッチフラグをクリア */
346
347  /* Configure TSRA
348  b7 OVFA = &0 オーバフローフラグをクリア
349  b6 Reserved
350  b5 ICFA5 = &0 インพุットキャプチャフラグをクリア
351  b4 ICFA4 = &0 インพุットキャプチャフラグをクリア
352  b3 ICFA3 = &0 インพุットキャプチャフラグをクリア
353  b2 ICFA2 = &0 インพุットキャプチャフラグをクリア
354  b1 ICFA1 = &0 インพุットキャプチャフラグをクリア
355  b0 ICFA0 = &0 インพุットキャプチャフラグをクリア */
356  ATUA.TSRA.BYTE &= 0x00; /* インพุットキャプチャフラグをクリア */
357
358  /* Configure TIERA
359  b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
360  b6 Reserved
361  b5 ICEA5 = 0 インพุットキャプチャ割り込み A5 要求の出力を禁止
362  b4 ICEA4 = 0 インพุットキャプチャ割り込み A4 要求の出力を禁止
363  b3 ICEA3 = 0 インพุットキャプチャ割り込み A3 要求の出力を禁止
364  b2 ICEA2 = 0 インพุットキャプチャ割り込み A2 要求の出力を禁止
365  b1 ICEA1 = 1 インพุットキャプチャ割り込み A1 要求の出力を許可
366  b0 ICEA0 = 0 インพุットキャプチャ割り込み A0 要求の出力を禁止 */
367  ATUA.TIERA.BYTE = 0x02; /* TIA01 のインพุットキャプチャ割り込みを許可 */
368
369  /* Configure TCNTB
370  b31:0 入力エッジ間計測カウンタ値 = 0 */
371  ATUB.TCNTB3.LONG = 0x00000000;
372
373  /* Configure TIORB
374  b7 LDSEL = 1 TCNTB2 のロードデータ、RLDB のロードデータの計算に LDB 値を用いる */
375  ATUB.TIORB.BIT.LDSEL=0x1;
376
377  /* Configure DCNTD
378  b31:8 タイマダウンカウンタ値
379  b7:0 Reserved */
380  ATUD.SUBBLOCKB[0].DCNTD[0] = puls_w; /* ダウンカウンタ 00 を 4 にセット */
381  ATUD.SUBBLOCKB[0].DCNTD[1] = puls_w; /* ダウンカウンタ 01 を 4 にセット */
382  ATUD.SUBBLOCKB[0].DCNTD[2] = puls_w; /* ダウンカウンタ 02 を 4 にセット */
383  } /* End of function CMIB6() */
384
385  /*****
386  * Function Name : CMID00
387  * Description   : コンペアマッチ割り込み(D0)

```

```

388  * Arguments      : none
389  * Return Value   : none
390  *****/
391  void CMID00(void)
392  {
393  /* Configure TSRD
394  b15:14 Reserved
395  b13 OVFDn = 0 オーバフローフラグをクリア
396  b12 OVFDn = 0 オーバフローフラグをクリア
397  b11 UDFDn3 = 0 アンダーフローフラグをクリア
398  b10 UDFDn2 = 0 アンダーフローフラグをクリア
399  b9  UDFDn1 = 0 アンダーフローフラグをクリア
400  b8  UDFDn0 = 0 アンダーフローフラグをクリア
401  b7  CMFADn3 = 0 コンペアマッチフラグをクリア
402  b6  CMFADn2 = 0 コンペアマッチフラグをクリア
403  b5  CMFADn1 = 0 コンペアマッチフラグをクリア
404  b4  CMFADn0 = 0 コンペアマッチフラグをクリア
405  b3  CMFBDn3 = 0 コンペアマッチフラグをクリア
406  b2  CMFBDn2 = 0 コンペアマッチフラグをクリア
407  b1  CMFBDn1 = 0 コンペアマッチフラグをクリア
408  b0  CMFBDn0 = 0 コンペアマッチフラグをクリア */
409  ATUD.SUBBLOCKB[0].TSRD.WORD &= 0x0000; /* コンペアマッチ、アンダフローフラグをクリア */
410  } /* End of function CMID00() */
411
412  /*****
413  * Function Name : ICIA1
414  * Description   : インプットキャプチャ割り込み
415  * Arguments     : none
416  * Return Value  : none
417  *****/
418  void ICIA1(void)
419  {
420  /* Configure TSRA
421  b7 OVFA = 0 オーバフローフラグをクリア
422  b6 Reserved
423  b5 ICFA5 = 0 インプットキャプチャフラグをクリア
424  b4 ICFA4 = 0 インプットキャプチャフラグをクリア
425  b3 ICFA3 = 0 インプットキャプチャフラグをクリア
426  b2 ICFA2 = 0 インプットキャプチャフラグをクリア
427  b1 ICFA1 = 0 インプットキャプチャフラグをクリア
428  b0 ICFA0 = 0 インプットキャプチャフラグをクリア*/
429  ATUA.TSRA.BYTE &= 0x00; /* インプットキャプチャフラグをクリア */
430
431  /* Configure TIERA
432  b7 OVEA = 0 オーバフロー割り込み A 要求の出力を禁止
433  b6 Reserved
434  b5 ICEA5 = 0 インプットキャプチャ割り込み A5 要求の出力を禁止
435  b4 ICEA4 = 0 インプットキャプチャ割り込み A4 要求の出力を禁止
436  b3 ICEA3 = 0 インプットキャプチャ割り込み A3 要求の出力を禁止
437  b2 ICEA2 = 0 インプットキャプチャ割り込み A2 要求の出力を禁止

```

アドバンスドタイマユニット-III (ATU-III) を使用したタイマ動作例

```
438  b1 ICEA1 = 0 インพุットキャプチャ割り込み A1 要求の出力を禁止
439  b0 ICEA0 = 0 インพุットキャプチャ割り込み A0 要求の出力を禁止 */
440  ATUA.TIERA.BYTE = 0x00; /* TIA01 のインพุットキャプチャ割り込みを禁止 */
441
442  /* Configure TIORB
443  b7 LDSEL = 0 TCNTB2 のロードデータ、RLDB のロードデータの計算に ICRB0 値を用いる */
444  ATUB.TIORB.BIT.LDSEL=0x0;
445  } /* End of function ICIA1() */
```

2.29.6 SH7253 グループでご使用の場合の変更点

本文 2.29 及びサンプルソフトは SH7254R グループ用に作成されたものです。SH7253 グループでご使用される場合は SH7254R グループのサンプルソフトを下記に従い変更してください。

変更内容

(1)製品ヘッダファイル iodef.h を SH7253 グループ用のものに差替えてください

(2)TIA01 端子が、SH7253 グループでは、PG13 端子（SH7254R グループでは、PE1 端子）に割り当てられているので、PFC(ピンファンクションコントローラ)の設定を変更してください。

具体的には

2.29.5 サンプルプログラムリストの 82 行目を

PORTG.CR4A.WORD=0x0010 に変更

SH7254R グループ : PE1CR を 0x0004 に設定 (変更前)

SH7253 グループ : PGCR4A を 0x0010 に設定 (変更後)

ホームページとサポート窓口

- ルネサス エレクトロニクスホームページ
<http://japan.renesas.com/>
- お問い合わせ先
<http://japan.renesas.com/inquiry>

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

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

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て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。



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口： <http://japan.renesas.com/inquiry>