
SH7239 グループ

R01AN0297JJ0100

Rev.1.00

初期設定例

2010.10.20

要旨

本アプリケーションノートは、SH7239 を起動する際に必要な初期設定例について説明しています。

動作確認デバイス

SH7239

目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. 参考プログラムリスト.....	13
4. 参考ドキュメント.....	26

1. はじめに

1.1 仕様

リセット解除後に、クロックパルス発振器（CPG）の初期設定を行います。

1.2 使用機能

- クロックパルス発振器（CPG）

1.3 適用条件

マイコン	SH7239（R5F72395ADFP）
電源電圧	3.3 V
動作周波数	内部クロック： 160 MHz バスクロック： 40 MHz 周辺クロック： 40 MHz
統合開発環境	ルネサス エレクトロニクス製 High-performance Embedded Workshop Ver.4.07.00
C コンパイラ	ルネサス エレクトロニクス製 SuperH RISC engine ファミリ C/C++コンパイラパッケージ Ver.9.03 Release 02
コンパイルオプション	High-performance Embedded Workshop でのデフォルト設定 (-cpu=sh2afpu -fpu=single -debug -gbr=auto -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1)

2. 応用例の説明

C言語で作成されたmain関数を実行するためには、リセット解除後に、CPGの設定など最低限のハードウェア初期化処理を行う初期設定プログラムが必要です。

SH7239の各アプリケーションノートでは、本アプリケーションノートで説明する参考プログラムを初期設定プログラムとして使用することを前提としています。

2.1 参考プログラムの説明

初期設定プログラムは、PowerON_Reset_PC関数が記述されたresetprg.cと、ハードウェア初期化処理を行うHardwareSetup関数が記述されたhwsetup.cなどの複数のソースファイルで構成されています。

初期設定プログラムを構成する主なソースファイルを以下に示します。

- resetprg.c
- hwsetup.c
- cpg.c

resetprg.cは、High-performance Embedded Workshopの自動生成ファイルを元に作成したもので、PowerON_Reset_PC関数が記述されています。PowerON_Reset_PC関数とは、リセット解除後最初に実行する関数で、vecttbl.cで定義するリセットベクタに関数の先頭アドレスが設定されています。

hwsetup.cは、PowerON_Reset_PC関数からコールされるHardwareSetup関数が記述されています。HardwareSetup関数では、CPGの設定処理を行うio_set_cpg関数をコールします。また、SDRAMの接続など、外部バスインタフェースを使用する場合は、io_set_cpg関数コール後、必要に応じてバスステートコントローラ(BSC)等の設定処理をHardwareSetup関数に追加してください。

cpg.cは、HardwareSetup関数からコールされるio_set_cpg関数が記述されています。io_set_cpg関数では、最初に周波数制御レジスタ(FRQCR)の設定を行います。FRQCRの設定は、内蔵RAM上のプログラムで行います(テクニカルアップデート「TN-SH7-A791A/J」参照)。参考プログラムでは、io_set_cpg関数の先頭で、FRQCR設定用のプログラムセクション(セクション名:PURAM)を内蔵ROMから内蔵RAMにコピーする処理(_seccpy関数)を実行した後、FRQCR設定処理(io_set_cpg_frqcr関数)を実行しています。FRQCR設定後は、MTUクロック周波数制御レジスタ(MCLKCR)およびADクロック周波数制御レジスタ(ACLKCR)を設定し、各内蔵周辺モジュールのモジュールスタンバイ機能を解除します。

図1～図4に、本応用例で使用する、上記ソースファイル内の初期設定プログラムの処理フローを示します。

【補足】スタック領域について

SH7239の内蔵RAM領域の内、ページ0およびページ1に対してはリード/ライトともにCPUが1サイクルでアクセスできます。本応用例では、この高速アクセス性能を考慮して、スタック領域を内蔵RAMのページ1(アドレス:H'FFF8 4000～H'FFF8 7FFF)の最後尾の領域に割り当てています。

スタック領域は「Sセクション」として割り当てており、High-performance Embedded Workshop上のメニュー「ビルド」→「SuperH RISC engine Standard Toolchain」→「最適化リンク」タブ内のカテゴリ「セクション」にて設定できます。高速アクセス性能を特に考慮しない場合はスタック領域を内蔵RAMのページ4またはページ5に割り当てることができます。スタック領域のセクション配置はシステムに応じて適宜変更してください。

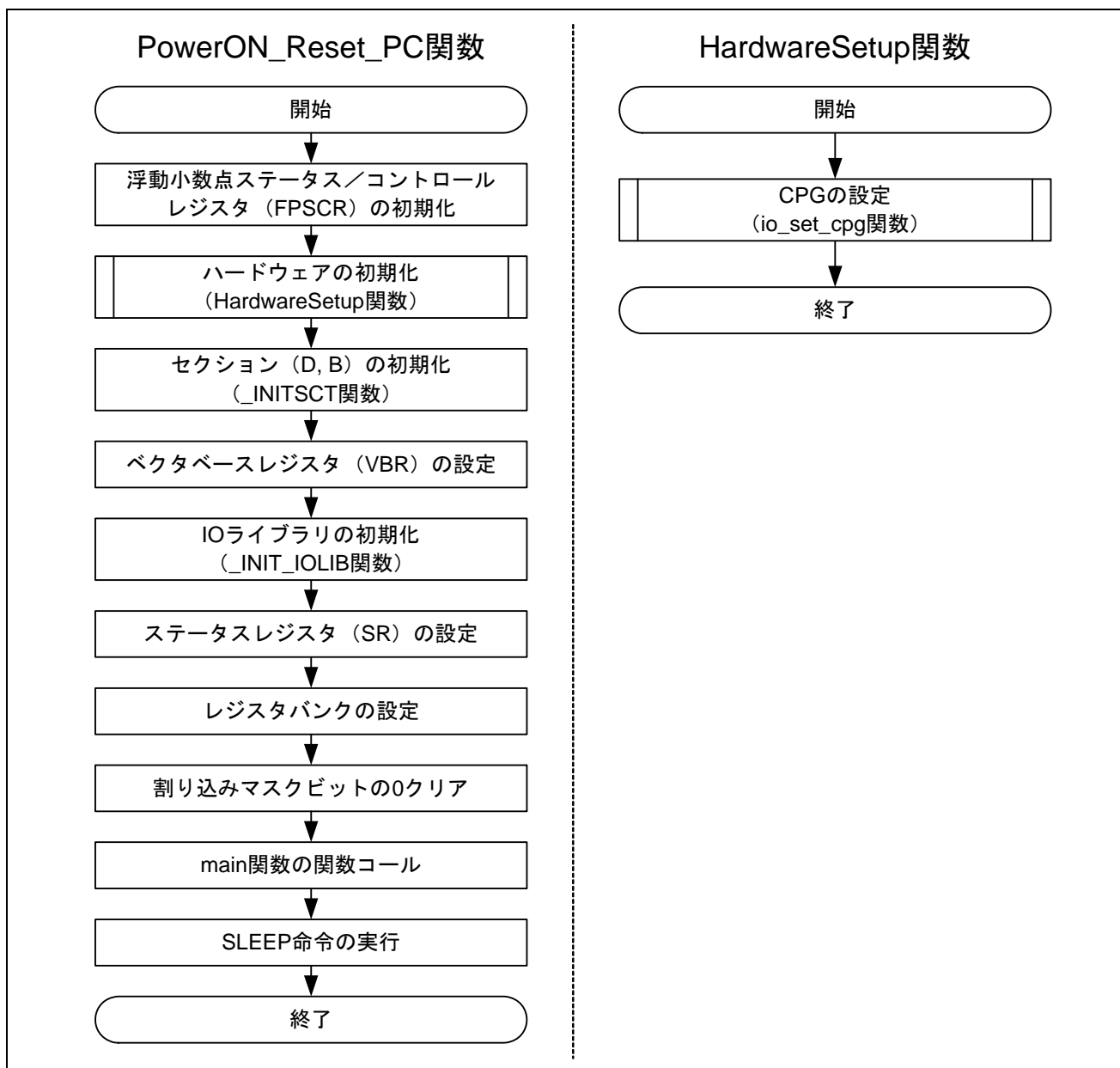


図1 PowerON_Reset_PC 関数および HardwareSetup 関数の処理フロー

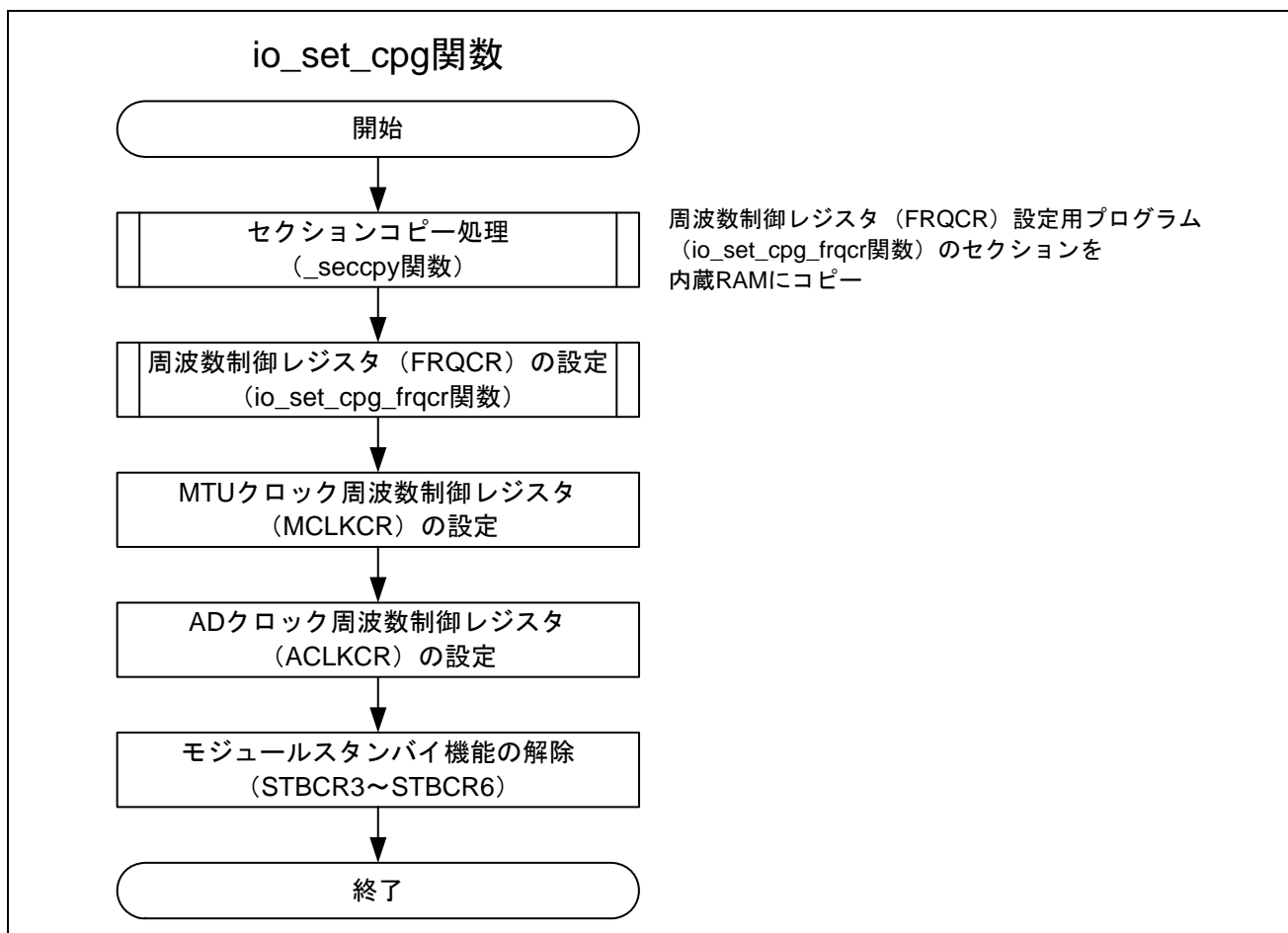


図2 CPG 設定処理 (io_set_cpg 関数) フロー

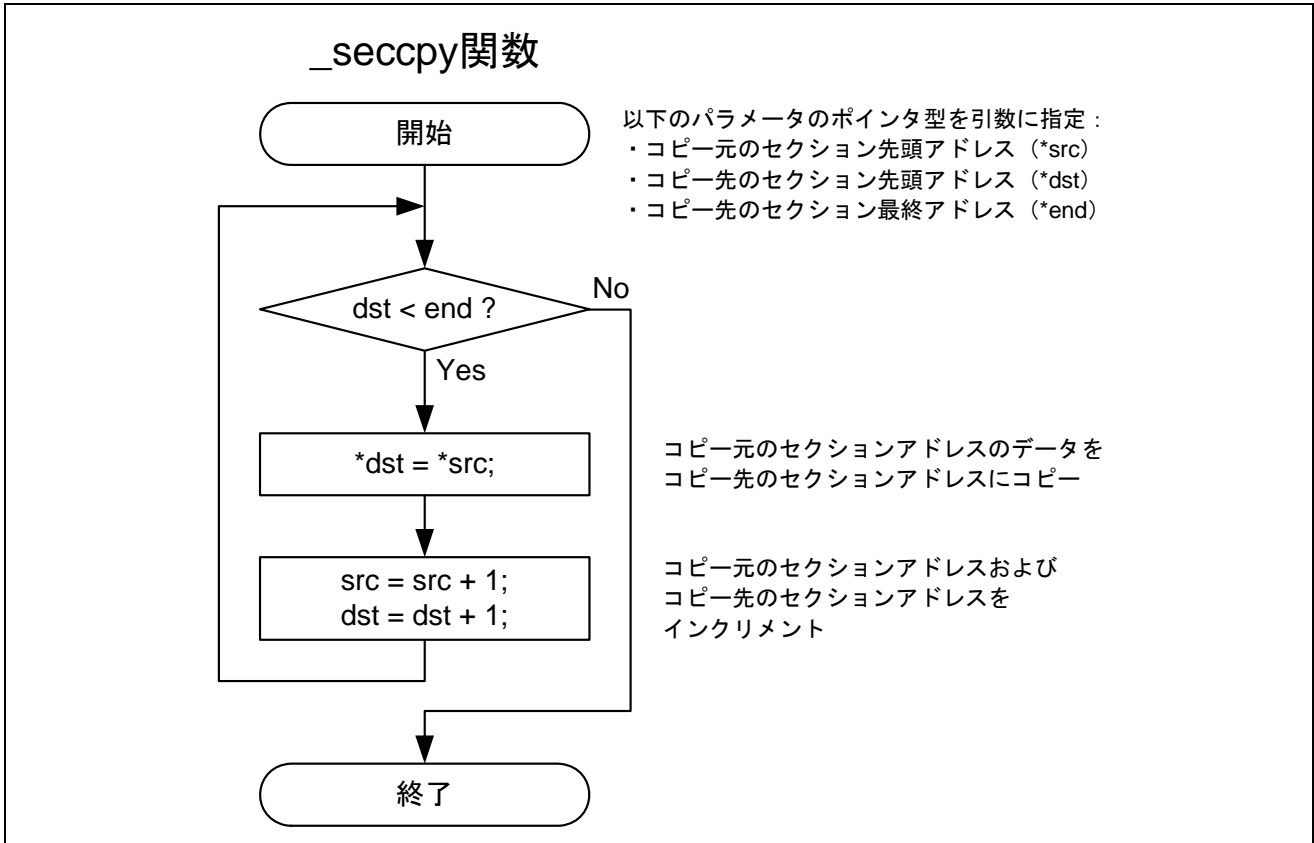


図3 セクションコピー処理 (_seccpy 関数) フロー

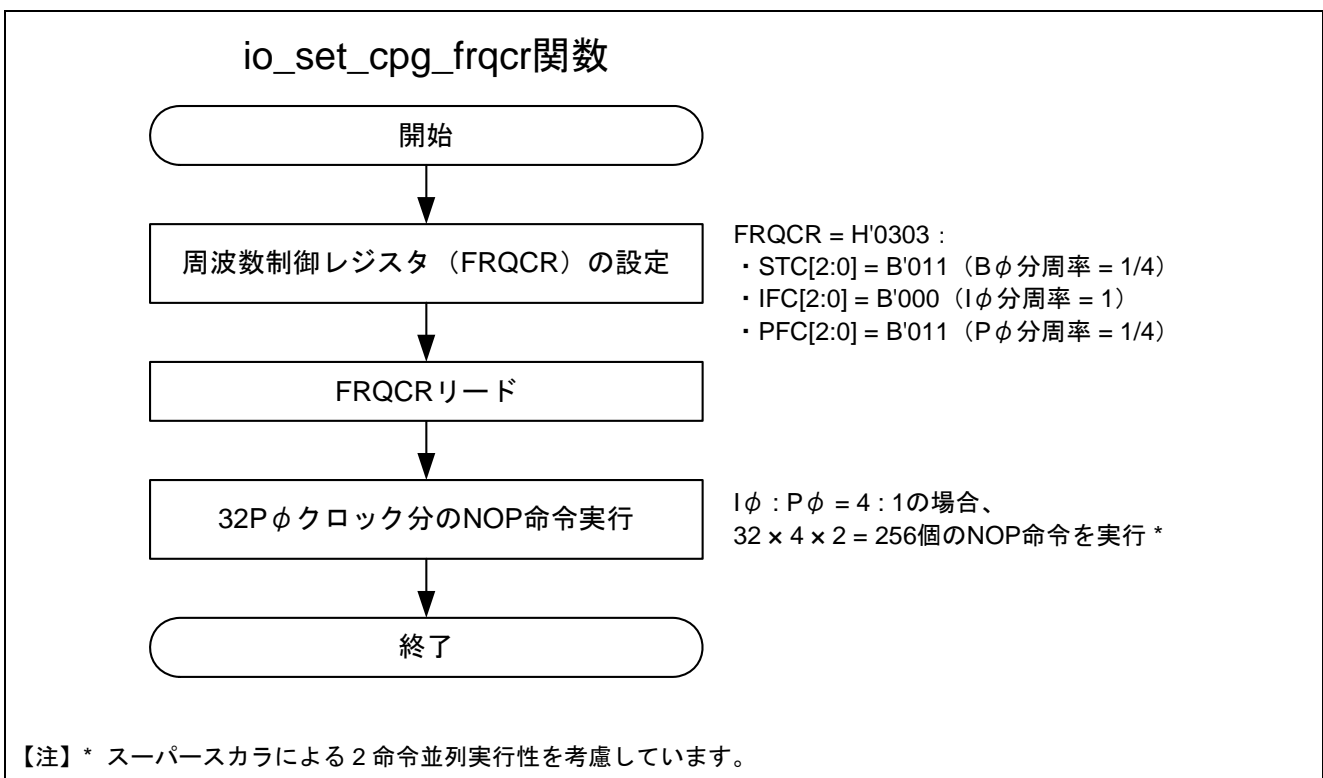


図4 FRQCR 設定処理 (io_set_cpg_frqcr 関数) フロー

2.2 使用機能の動作概要

CPG は、内部クロック (I ϕ)、バスクロック (B ϕ)、周辺クロック (P ϕ)、MTU クロック (M ϕ)、および AD クロック (A ϕ) の生成と、低消費電力モードの制御を行います。

表 1 に CPG の概要を示します。また、図 5 に CPG のブロック図を示します。

表1 CPG の概要

項目	内容
生成クロック	<ul style="list-style-type: none">内部クロック (Iϕ) : CPU で使用バスクロック (Bϕ) : 外部バスインタフェースで使用周辺クロック (Pϕ) : 内蔵周辺モジュールで使用MTU クロック (Mϕ) : MTU2/MTU2S モジュールで使用AD クロック (Aϕ) : ADC モジュールで使用
周波数変更機能	<ul style="list-style-type: none">CPG 内部の PLL (Phase Locked Loop) 回路や分周回路により、生成クロックの周波数をそれぞれ独立して設定可能周波数の変更は、周波数制御レジスタ (FRQCR, MCLKCR, ACLKCR) の設定により、ソフトウェアで行う
低消費電力モードの制御	スリープモード、スタンバイモードでのクロック停止、またモジュールスタンバイ機能での指定モジュールの停止が可能

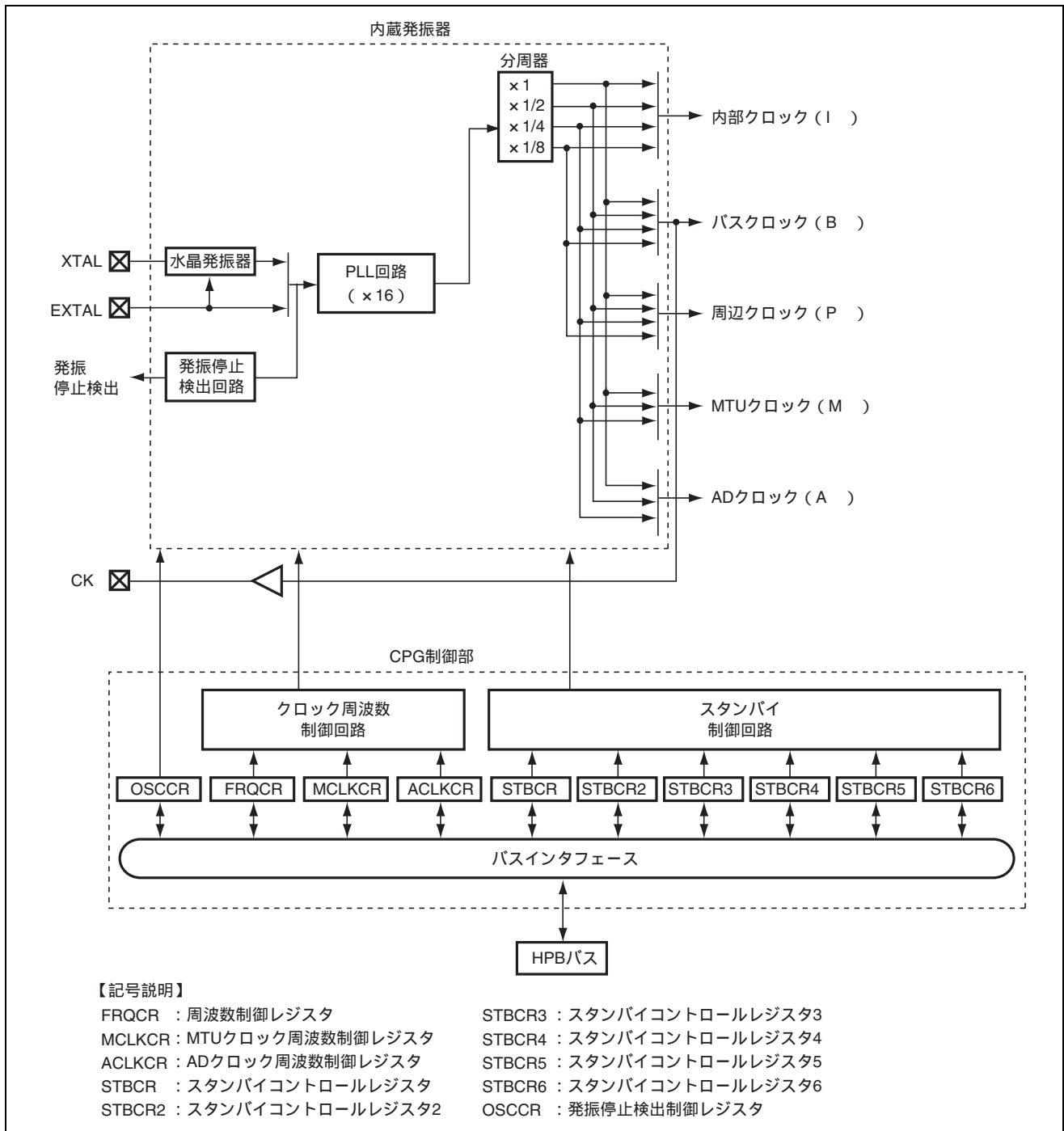


図5 CPGのブロック図

2.3 使用機能の設定手順

図 6に、CPGの設定フロー例を示します。リセット解除後、各内蔵周辺モジュールはモジュールスタンバイモードとなっています。参考プログラムでは、周波数制御レジスタ (FRQCR)、MTUクロック周波数制御レジスタ (MCLKCR)、およびADクロック周波数制御レジスタ (ACLKCR) の設定後に各内蔵周辺モジュールのモジュールスタンバイ機能を解除します。

各レジスタの詳細については、「SH7239 グループ、SH7237 グループ ユーザーズマニュアル ハードウェア編 クロックパルス発振器 (CPG)」を参照してください。

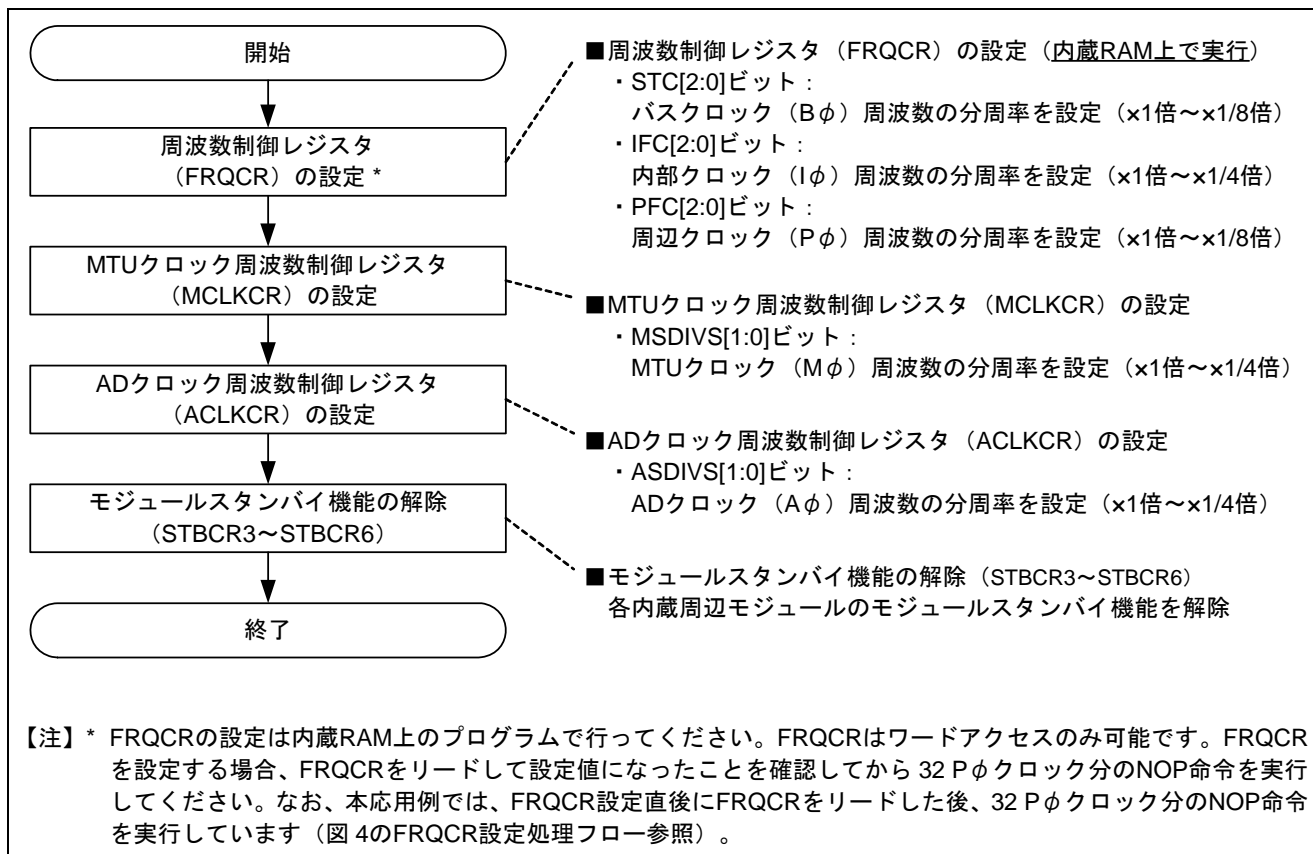


図6 CPG の設定フロー例

2.4 参考プログラムの設定内容

表 2に参考プログラムにおける使用機能の設定内容を、また 表 3および 表 4に参考プログラムにおける使用機能のレジスタ設定内容を示します。

表2 参考プログラムにおける使用機能の設定内容

モジュール	設定内容
浮動小数点ユニット (FPU)	<ul style="list-style-type: none"> ● 精度モード 浮動小数点命令を単精度で実行 ● 丸めモード 0 方向への丸め
クロックパルス発振器 (CPG)	<ul style="list-style-type: none"> ● 生成クロック周波数 (入力クロック=10 MHz) : <ul style="list-style-type: none"> — 内部クロック : 160 MHz — バスクロック : 40 MHz — 周辺クロック : 40 MHz — MTU クロック : 80 MHz — AD クロック : 40 MHz ● モジュールスタンバイ機能を解除するモジュール : MTU2S, MTU2, ADC0, ADC1, ADC2, CMT, SCIO, SCI1, SCI2, SCIF3, RSPI, RCAN-ET

表3 参考プログラムにおける使用機能のレジスタ設定内容 (1)

レジスタ	アドレス	設定値	設定内容
周波数制御レジスタ (FRQCR)	H'FFFE 0010	H'0303	<ul style="list-style-type: none"> ● STC[2:0] = "B'011" : バスクロック周波数 (Bϕ) の分周率 = 1/4 倍 ● IFC[2:0] = "B'000" : 内部クロック周波数 (Iϕ) の分周率 = 1 倍 ● PFC[2:0] = "B'011" : 周辺クロック周波数 (Pϕ) の分周率 = 1/4 倍
MTU クロック周波数制御レジスタ (MCLKCR)	H'FFFE 0410	H'41	<ul style="list-style-type: none"> ● MSDIVS[1:0] = "B'01" : MTU クロック周波数 (Mϕ) の分周率 = 1/2 倍
AD クロック周波数制御レジスタ (ACLKCR)	H'FFFE 0414	H'43	<ul style="list-style-type: none"> ● ASDIVS[1:0] = "B'11" : AD クロック周波数 (Aϕ) の分周率 = 1/4 倍

表4 参考プログラムにおける使用機能のレジスタ設定内容 (2)

レジスタ	アドレス	設定値	設定内容
スタンバイコントロール レジスタ 3 (STBCR3)	H'FFFE 0408	H'1A	<ul style="list-style-type: none"> • HIZ = "0" : ソフトウェアスタンバイモード時に、 端子状態を保持 • MSTP36 = "0" : MTU2S は動作 • MSTP35 = "0" : MTU2 は動作 • MSTP32 = "0" : ADC0 は動作 • MSTP30 = "0" : フラッシュメモリは動作
スタンバイコントロール レジスタ 4 (STBCR4)	H'FFFE 040C	H'E3	<ul style="list-style-type: none"> • MSTP44 = "0" : SCIF3 は動作 • MSTP42 = "0" : CMT は動作
スタンバイコントロール レジスタ 5 (STBCR5)	H'FFFE 0418	H'18	<ul style="list-style-type: none"> • MSTP57 = "0" : SCI0 は動作 • MSTP56 = "0" : SCI1 は動作 • MSTP55 = "0" : SCI2 は動作 • MSTP52 = "0" : ADC1 は動作 • MSTP51 = "0" : ADC2 は動作 • MSTP50 = "0" : RSPI は動作
スタンバイコントロール レジスタ 6 (STBCR6)	H'FFFE 041C	H'CF	<ul style="list-style-type: none"> • MSTP64 = "0" : RCAN-ET は動作

【補足】ROM化支援オプションについて

本応用例では、FRQCR 設定処理 (io_set_cpg_frqcr 関数) 用のプログラムセクションを、内蔵 ROM から内蔵 RAM にコピーしています。このようなセクションコピー処理を追加する場合、C コンパイラの最適化リンカージェディタにて ROM 化支援オプションを設定する必要があります。

ROM 化支援オプションを設定する場合、High-performance Embedded Workshop 上のメニュー「ビルド」→「SuperH RISC engine Standard Toolchain」→「最適化リンカ」タブ内のカテゴリ「出力」にて、オプション項目「ROM から RAM へマップするセクション」を選択します。ここで「追加」ボタンをクリックし、「ROM セクション」にはコピー元のセクションを指定し、「RAM セクション」にはコピー先のセクションを指定します。このとき、あらかじめ「最適化リンカ」タブ内のカテゴリ「セクション」にて、コピー元およびコピー先のセクション配置場所を設定しておきます。本応用例では、FRQCR 設定処理用のプログラムセクションを "PURAM" とし、そのコピー先の RAM セクションを "RPURAM" としてそれぞれ配置場所を設定しています。

図 7 に、ROM 化支援オプション設定例を示します。ROM 化支援オプションの詳細は、「SuperH C/C++コンパイラパッケージ V.9.04 ユーザーズマニュアル」を参照してください。

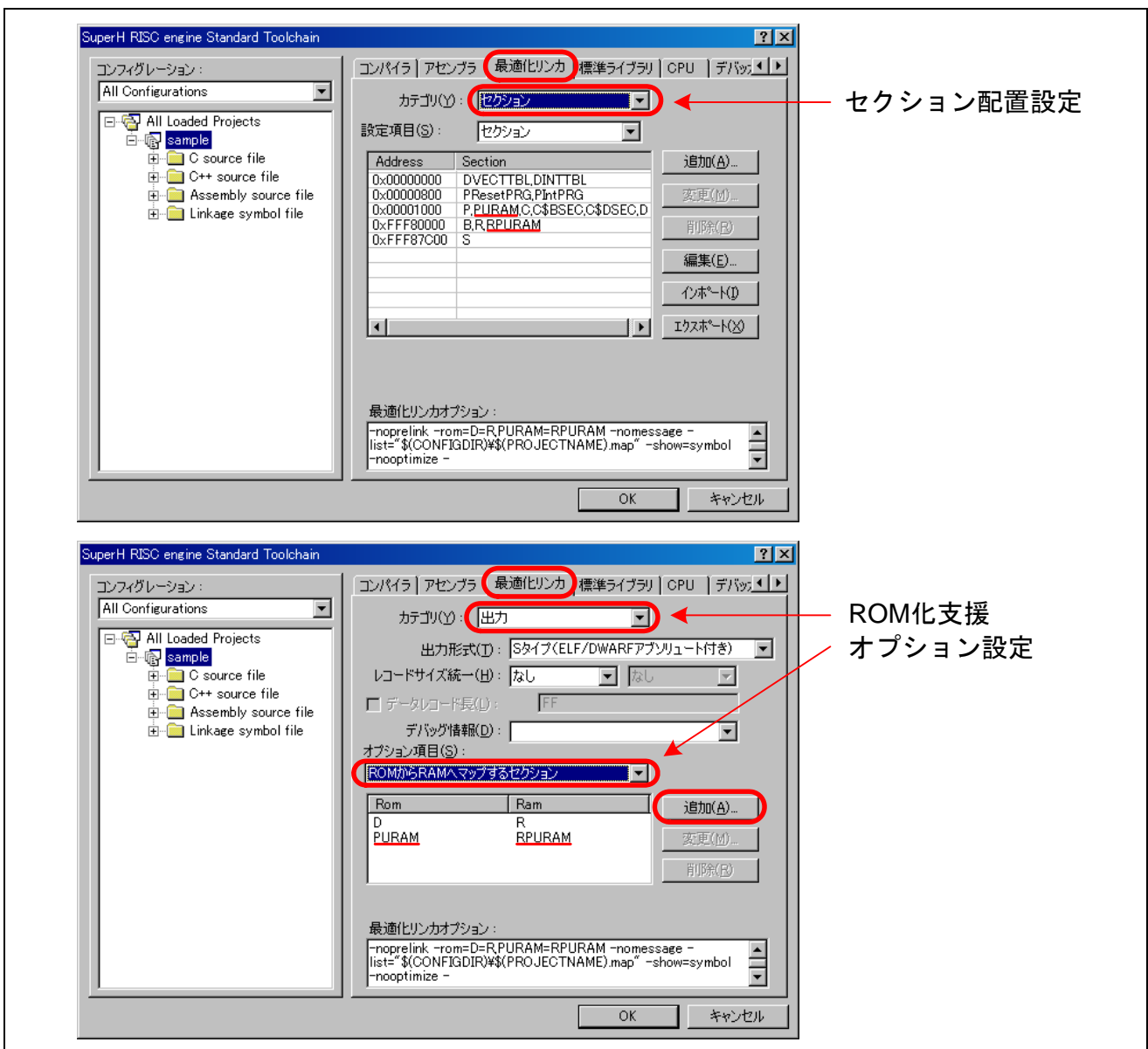


図7 ROM化支援オプション設定例

3. 参考プログラムリスト

3.1 サンプルプログラムリスト "resetprg.c" (1)

```
1  /*****
2  *   DISCLAIMER
3  *
4  *   This software is supplied by Renesas Electronics Corporation and is only
5  *   intended for use with Renesas products. No other uses are authorized.
6  *
7  *   This software is owned by Renesas Electronics Corporation and is protected under
8  *   all applicable laws, including copyright laws.
9  *
10 *   THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *   REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *   INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *   PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *   DISCLAIMED.
15 *
16 *   TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *   ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *   FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *   FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *   AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *   Renesas reserves the right, without notice, to make changes to this
23 *   software and to discontinue the availability of this software.
24 *   By using this software, you agree to the additional terms and
25 *   conditions found by accessing the following link:
26 *   http://www.renesas.com/disclaimer
27 *****/
28 *   Copyright (C) 2010 Renesas Electronics Corporation. All rights reserved.
29 *****/
30 /*"FILE COMMENT"***** Technical reference data *****
31 *   System Name : SH7239 Sample Program
32 *   File Name   : resetprg.c
33 *   Abstract    : SH7239 Initial Setting
34 *   Version     : 1.01.00
35 *   Device      : SH7239
36 *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
37 *                : C/C++ compiler package for the SuperH RISC engine family
38 *                :                               (Ver.9.03 Release02).
39 *   OS          : None
40 *   H/W Platform: R0K572390 (CPU board)
41 *   Description :
42 *****/
43 *   History     : Aug.20,2010 Ver.1.00.00
44 *                : Oct.20,2010 Ver.1.01.00 Add the IO library initialization
45 * "FILE COMMENT END"*****
```

3.2 サンプルプログラムリスト "resetprg.c" (2)

```
46  #include <machine.h>
47  #include <_h_c_lib.h>
48  #include "stacksct.h"
49  #include "iodefine.h"
50
51  /* ==== Macro definition ==== */
52  #define FPSCR_Init  0x00040001
53  #define SR_Init     0x000000f0
54  #define INT_OFFSET  0x10
55
56  /* ==== Prototype declaration ==== */
57  void PowerON_Reset_PC(void);
58  void Manual_Reset_PC(void);
59
60  /* ==== External reference declaration ==== */
61  /* ---- Function prototype ---- */
62  extern void main(void);
63  extern void HardwareSetup(void);
64  /* ---- Global variable ---- */
65  extern unsigned int INT_Vectors;
66
67  /* ==== Section name changed to ResetPRG ==== */
68  #pragma section ResetPRG
69
70  /* ==== Entry function specified ==== */
71  #pragma entry PowerON_Reset_PC
72
73  /*"FUNC COMMENT"*****
74  * ID          :
75  * Outline     : CPU initialization
76  *-----
77  * Include     : <machine.h>, <_h_c_lib.h>, and "iodefine.h"
78  *-----
79  * Declaration : void PowerON_Reset_PC(void);
80  *-----
81  * Description : Executes the CPU initialization processing to register
82  *              : the power-on reset vector to the exception vector table.
83  *-----
84  * Argument    : void
85  *-----
86  * Return Value : void
87  *-----
88  * Note        : This function is executed first after power-on reset.
89  *"FUNC COMMENT END"*****
```

3.3 サンプルプログラムリスト "resetprg.c" (3)

```
90 void PowerON_Reset_PC(void)
91 {
92     /* ==== Floating Point Status/Control Register setting ==== */
93     set_fpscr(FPSCR_Init);
94
95     /* ==== Hardware initialization ==== */
96     HardwareSetup();          /* HardwareSetup function */
97
98     /* ==== Sections initialization ==== */
99     _INITSCT();
100
101     /* ==== Vector Base Register setting ==== */
102     set_vbr((void *)((char *)&INT_Vectors - INT_OFFSET));
103
104     /* ==== IO library initialization ==== */
105     _INIT_IOLIB();
106
107     /* ==== Status Register setting ==== */
108     set_cr(SR_Init);
109     nop();
110
111     /* ==== Bunk Number Register setting ==== */
112     INTC.IBNR.BIT.BE = 1;     /* Use of register banks enabled for all */
113                               /* interrupts except NMI and user break */
114
115     /* ==== Interrupt mask bits clear ==== */
116     set_imask(0);
117
118     /* ==== Main function call ==== */
119     main();
120
121     /* ==== Sleep instruction execution ==== */
122     sleep();
123 }
124
```

3.4 サンプルプログラムリスト "resetprg.c" (4)

```
125  /*"FUNC COMMENT"*****  
126  * ID      :  
127  * Outline : Manual reset processing  
128  *-----  
129  * Include :  
130  *-----  
131  * Declaration : void Manual_Reset_PC(void);  
132  *-----  
133  * Description : Registers the manual reset vector to the exception vector  
134  *             : table.  
135  *-----  
136  * Argument  : void  
137  *-----  
138  * Return Value : void  
139  *-----  
140  * Note      : This sample does not describe the processing content at all.  
141  *           : Add the program in this function as needed.  
142  /*"FUNC COMMENT END"*****/  
143 void Manual_Reset_PC(void)  
144 {  
145     /* NOP */  
146 }  
147  
148 /* END of File */
```


3.5 サンプルプログラムリスト "hwsetup.c" (1)

```
1  /*****
2  *   DISCLAIMER
3  *
4  *   This software is supplied by Renesas Electronics Corporation and is only
5  *   intended for use with Renesas products. No other uses are authorized.
6  *
7  *   This software is owned by Renesas Electronics Corporation and is protected under
8  *   all applicable laws, including copyright laws.
9  *
10 *   THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *   REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *   INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *   PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *   DISCLAIMED.
15 *
16 *   TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *   ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *   FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *   FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *   AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *   Renesas reserves the right, without notice, to make changes to this
23 *   software and to discontinue the availability of this software.
24 *   By using this software, you agree to the additional terms and
25 *   conditions found by accessing the following link:
26 *   http://www.renesas.com/disclaimer
27 *****/
28 *   Copyright (C) 2010 Renesas Electronics Corporation. All rights reserved.
29 *****/
30 /*"FILE COMMENT"***** Technical reference data *****
31 *   System Name : SH7239 Sample Program
32 *   File Name   : hwsetup.c
33 *   Abstract    : Hardware Function Initial Setting
34 *   Version     : 1.00.00
35 *   Device      : SH7239
36 *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
37 *               : C/C++ compiler package for the SuperH RISC engine family
38 *               :                               (Ver.9.03 Release02).
39 *   OS          : None
40 *   H/W Platform: R0K572390 (CPU board)
41 *   Description :
42 *****/
43 *   History     : Aug.20,2010 Ver.1.00.00
44 /*"FILE COMMENT END"*****
45 #include "iodefine.h"
46
47 /* ==== Prototype declaration ==== */
48 void HardwareSetup(void);
49
```

3.6 サンプルプログラムリスト "hwsetup.c" (2)

```
50  /* ==== External reference ==== */
51  /* ---- Function prototype ---- */
52  extern void io_set_cpg(void);
53
54  /*"FUNC COMMENT"*****
55  * ID          :
56  * Outline     : Hardware initialization
57  *-----
58  * Include     :
59  *-----
60  * Declaration : void HardwareSetup(void);
61  *-----
62  * Description : Initializes the hardware function.
63  *-----
64  * Argument    : void
65  *-----
66  * Return Value : void
67  *-----
68  * Note        : None
69  *"FUNC COMMENT END"*****/
70  void HardwareSetup(void)
71  {
72      /* ==== CPG setting ==== */
73      io_set_cpg();
74  }
75
76  /* End of File */
```

3.7 サンプルプログラムリスト "cpg.c" (1)

```
1  /*****
2  *   DISCLAIMER
3  *
4  *   This software is supplied by Renesas Electronics Corporation and is only
5  *   intended for use with Renesas products. No other uses are authorized.
6  *
7  *   This software is owned by Renesas Electronics Corporation and is protected under
8  *   all applicable laws, including copyright laws.
9  *
10 *   THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *   REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *   INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *   PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *   DISCLAIMED.
15 *
16 *   TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *   ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *   FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *   FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *   AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *   Renesas reserves the right, without notice, to make changes to this
23 *   software and to discontinue the availability of this software.
24 *   By using this software, you agree to the additional terms and
25 *   conditions found by accessing the following link:
26 *   http://www.renesas.com/disclaimer
27 *****/
28 *   Copyright (C) 2010 Renesas Electronics Corporation. All rights reserved.
29 *****/
30 /*"FILE COMMENT"***** Technical reference data *****/
31 *   System Name : SH7239 Sample Program
32 *   File Name   : cpg.c
33 *   Abstract    : CPG Setting Processing
34 *   Version     : 1.01.00
35 *   Device      : SH7239
36 *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
37 *               : C/C++ compiler package for the SuperH RISC engine family
38 *               :                               (Ver.9.03 Release02).
39 *   OS          : None
40 *   H/W Platform: R0K572390 (CPU board)
41 *   Description :
42 *****/
43 *   History     : Aug.20,2010 Ver.1.00.00
44 *               : Oct.20,2010 Ver.1.01.00 - Divide the FRQCR setting into
45 *               :                               subroutine "io_set_cpg_frqcr"
46 *               :                               which is allocated to on-chip RAM
47 *               :                               - Add the processing of section copy
48 *               :                               function "_seccpy"
49 *               :                               - Modify comment
50 * "FILE COMMENT END"*****/
```

3.8 サンプルプログラムリスト "cpg.c" (2)

```

51  #include <machine.h>
52  #include "iodefine.h"
53  #include "cpumodel.h"
54
55  /* ==== Prototype declaration ==== */
56  void io_set_cpg(void);
57  void io_set_cpg_frqcr(void);
58  static void _seccpy(unsigned long *src, unsigned long *dst, unsigned long *end);
59
60  /*"FUNC COMMENT"*****
61  * ID          :
62  * Outline     : CPG setting
63  *-----
64  * Include     : "iodefine.h"
65  *-----
66  * Declaration : void io_set_cpg(void);
67  *-----
68  * Description : Initializes the clock pulse generator (CPG) as follows:
69  *              : SH7239A: I-clock = 160MHz, B-clock = 40MHz, P-clock = 40MHz,
70  *              :              M-clock = 80MHz, and A-clock = 40MHz.
71  *              : SH7239B: I-clock = 100MHz, B-clock = 50MHz, P-clock = 50MHz,
72  *              :              M-clock = 100MHz, and A-clock = 50MHz.
73  *              : And then supplies clock to all peripheral modules.
74  *-----
75  * Argument    : void
76  *-----
77  * Return value : void
78  *-----
79  * Note        : This function is an example of CPG setting at the input clock
80  *              : of 10MHz/12.5MHz.
81  *"FUNC COMMENT END"*****/
82  void io_set_cpg(void)
83  {
84      /* ==== CPG setting ==== */
85      /* ---- Program section initialization for FRQCR setting ---- */
86      _seccpy((unsigned long *)__sectop("PURAM"), (unsigned long *)__sectop("RPURAM"),
87             (unsigned long *)__secend("RPURAM"));
88      /* Program section copying from "PURAM" to on-chip RAM */
89      /* ---- FRQCR setting (Running on-chip RAM) ---- */
90      io_set_cpg_frqcr();          /* Clock-in = 10MHz/ 12.5MHz: */
91                                 /* I-clock = 160MHz/100MHz, */
92                                 /* B-clock = 40MHz/ 50MHz, */
93                                 /* P-clock = 40MHz/ 50MHz */
94      /* ---- MCLKCR setting ---- */
95      CPG.MCLKCR.BYTE = 0x41;     /* M-clock = 80MHz/100MHz */
96      /* ---- ACLKCR setting ---- */
97      CPG.ACLKCR.BYTE = 0x43;    /* A-clock = 40MHz/ 50MHz */
98
99

```

3.9 サンプルプログラムリスト "cpg.c" (3)

```

100  /* ==== Module standby clear ==== */
101  /* ---- STBCR3 setting ---- */
102  STB.CR3.BYTE = 0x1a; /* HIZ,MTU2S,MTU2,Reserve(1),          */
103                    /* Reserve(1),ADC0,Reserve(1),FLASH      */
104  /* ---- STBCR4 setting ---- */
105  STB.CR4.BYTE = 0xe3; /* Reserve(1),Reserve(1),Reserve(1),SCIF3,          */
106                    /* Reserve(0),CMT,Reserve(1),Reserve(1)          */
107  /* ---- STBCR5 setting ---- */
108  STB.CR5.BYTE = 0x18; /* SCI0,SCI1,SCI2,Reserve(1),          */
109                    /* Reserve(1),ADC1,ADC2,RSPI          */
110  /* ---- STBCR6 setting ---- */
111  STB.CR6.BYTE = 0xcf; /* Reserve(1),Reserve(1),Reserve(0),RCAN-ET,          */
112                    /* Reserve(1),Reserve(1),Reserve(1),Reserve(1) */
113  }
114
115  /* ==== Section name changed to URAM ==== */
116  #pragma section URAM
117  /*"FUNC COMMENT"*****
118  * ID          :
119  * Outline     : FRQCR register setting
120  *-----
121  * Include     : <machine.h> and "iodefine.h"
122  *-----
123  * Declaration : void io_set_cpg_frqcr(void);
124  *-----
125  * Description : Initializes the clock pulse generator (CPG) as follows:
126  *              : SH7239A: I-clock = 160MHz, B-clock = 40MHz, P-clock = 40MHz.
127  *              : SH7239B: I-clock = 100MHz, B-clock = 50MHz, P-clock = 50MHz.
128  *-----
129  * Argument    : void
130  *-----
131  * Return Value : void
132  *-----
133  * Note        : - This function needs to be run on internal RAM.
134  *              : - This function is also an example of CPG setting at the
135  *              :   input clock of 10MHz/12.5MHz.
136  *"FUNC COMMENT END"*****
137  void io_set_cpg_frqcr(void)
138  {
139  #ifndef CPU_MODEL_SH7239B /* When using SH7239A */
140     CPG.FRQCR.WORD = 0x0303; /* Clock-in = 10MHz: */
141                            /* I-clock = 160MHz, */
142                            /* B-clock = 40MHz, */
143                            /* P-clock = 40MHz */
144     CPG.FRQCR.WORD; /* FRQCR readout */
145
146     /* ---- 256 NOPs for 32 x P-clock (I:P = 4:1) ---- */
147     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
148     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
149     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
150     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();

```

3.10 サンプルプログラムリスト "cpg.c" (4)

```
151     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
152     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
153     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
154     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
155     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
156     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
157     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
158     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
159     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
160     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
161     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
162     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
163     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
164     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
165     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
166     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
167     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
168     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
169     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
170     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
171     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
172     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
173     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
174     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
175     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
176     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
177     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
178     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
179
180     #else                                     /* When using SH7239B */
181         CPG.FRQCR.WORD = 0x0313;             /* Clock-in = 12.5MHz: */
182                                             /* I-clock = 100MHz, */
183                                             /* B-clock = 50MHz, */
184                                             /* P-clock = 50MHz */
185         CPG.FRQCR.WORD;                     /* FRQCR readout */
186
187     /* ---- 128 NOPs for 32 x P-clock (I:P = 2:1) ---- */
188     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
189     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
190     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
191     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
192     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
193     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
194     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
195     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
196     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
197     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
198     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
199     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
200     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
201     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
```

3.11 サンプルプログラムリスト "cpg.c" (5)

```
202     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
203     nop(); nop(); nop(); nop(); nop(); nop(); nop(); nop();
204 #endif
205 }
206 #pragma section      /* End of "URAM" section */
207
208 /*"FUNC COMMENT"*****
209 * ID          :
210 * Outline     : Section copy function
211 *-----
212 * Include     :
213 *-----
214 * Declaration : static void _seccpy(unsigned long *src, unsigned long *dst,
215 *                               :               unsigned long *end);
216 *-----
217 * Description : Copies a source section to specified target.
218 *-----
219 * Argument    : unsigned long *src ; Initial address of source section
220 *              : unsigned long *dst ; Initial address of target section
221 *              : unsigned long *end ; Final address of target section
222 *-----
223 * Return Value : void
224 *-----
225 * Note        : None
226 *"FUNC COMMENT END"*****/
227 static void _seccpy(unsigned long *src, unsigned long *dst, unsigned long *end)
228 {
229     while(dst < end){
230         *dst++ = *src++;
231     }
232 }
233
234 /* End of File */
```

3.12 サンプルプログラムリスト "vecttbl.c" (1)

```
1  /*****
2  *   DISCLAIMER
3  *
4  *   This software is supplied by Renesas Electronics Corporation and is only
5  *   intended for use with Renesas products. No other uses are authorized.
6  *
7  *   This software is owned by Renesas Electronics Corporation and is protected under
8  *   all applicable laws, including copyright laws.
9  *
10 *   THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *   REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *   INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *   PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *   DISCLAIMED.
15 *
16 *   TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *   ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *   FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *   FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *   AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *   Renesas reserves the right, without notice, to make changes to this
23 *   software and to discontinue the availability of this software.
24 *   By using this software, you agree to the additional terms and
25 *   conditions found by accessing the following link:
26 *   http://www.renesas.com/disclaimer
27 *****/
28 *   Copyright (C) 2010 Renesas Electronics Corporation. All rights reserved.
29 *****/
30 /*"FILE COMMENT"***** Technical reference data *****/
31 *   System Name : SH7239 Sample Program
32 *   File Name   : vecttbl.c
33 *   Abstract    : Initialization for Vector Table
34 *   Version     : 1.00.00
35 *   Device      : SH7239
36 *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
37 *               : C/C++ compiler package for the SuperH RISC engine family
38 *               :                               (Ver.9.03 Release02).
39 *   OS          : None
40 *   H/W Platform: R0K572390 (CPU board)
41 *   Description :
42 *****/
43 *   History     : Aug.20,2010 Ver.1.00.00
44 /*"FILE COMMENT END"*****/
45 #include "vect.h"
46
47 #pragma section VECTTBL
48 void *RESET_Vectors[] = {
49 // <<VECTOR DATA START (POWER ON RESET)>>
50 // 0 Power On Reset PC
51     (void *)PowerON_Reset_PC,
```


3.13 サンプルプログラムリスト "vecttbl.c" (2)

```
52 // <<VECTOR DATA END (POWER ON RESET)>>
53 // 1 Power On Reset SP
54   __secend("S"),
55 // <<VECTOR DATA START (MANUAL RESET)>>
56 // 2 Manual Reset PC
57   (void *)Manual_Reset_PC,
58 // <<VECTOR DATA END (MANUAL RESET)>>
59 // 3 Manual Reset SP
60   __secend("S")
61 };
62
63 #pragma section INTTBL
64 void *INT_Vectors[] = {
65 // 4 Illegal code
66   (void *)INT_Illegal_code,
67
68   ...
69
70 // 255 SCIF SCIF3 TXI3
71   (void *)INT_SCIF_SCIF3_TXI3,
72 // xx Reserved
73   (void *)Dummy
74 };
75
76 /* End of File */
```

4. 参考ドキュメント

- ソフトウェアマニュアル
SH-2A、SH2A-FPU ソフトウェアマニュアル Rev.3.00
(最新版をルネサス エレクトロニクス ホームページから入手してください。)
- ハードウェアマニュアル
SH7239 グループ、SH7237 グループ ユーザーズマニュアル：ハードウェア編 Rev.1.00
(最新版をルネサス エレクトロニクス ホームページから入手してください。)
- テクニカルアップデート
SH7239 グループ、SH7237 グループユーザーズマニュアルハードウェア編の誤記訂正 (TN-SH7-A791A/J)
(最新版をルネサス エレクトロニクス ホームページから入手してください。)
- 開発環境マニュアル
SuperH C/C++コンパイラパッケージ V.9.04 ユーザーズマニュアル Rev.1.00
(最新版をルネサス エレクトロニクス ホームページから入手してください。)

ホームページとサポート窓口

ルネサス 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。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.10.20	—	初版発行

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っていると、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、海中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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

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



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>