

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

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

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

2010年4月1日

ルネサスエレクトロニクス株式会社

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

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

## ご注意書き

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

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

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

# SH7619 グループ

## SH7619 初期設定例

---

### 要旨

この資料は、SH7619 の起動時に必要な設定項目の例を示します。

### 動作確認デバイス

SH7619

### 目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. ユーザプログラム領域の RAM 転送設定 .....	6
4. 参考プログラムリスト .....	8
5. 参考ドキュメント.....	19

## 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）、バスステートコントローラ（BSC）、ピンファンクションコントローラ（PFC）、およびキャッシュ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）
- バスステートコントローラ（BSC）
- ピンファンクションコントローラ（PFC）
- キャッシュ

### 1.3 適用条件

- マイコン: SH7619 (R4S76190)
- 動作周波数: 内部クロック 125MHz  
バスクロック 62.5MHz  
周辺クロック 31.25MHz
- 統合開発環境: ルネサステクノロジ製 High-performance Embedded Workshop Ver.4.03.00
- C コンパイラ: ルネサステクノロジ製  
SuperH RISC engine ファミリ C/C++コンパイラパッケージ Ver.9.01 Release01
- コンパイルオプション: High-performance Embedded Workshop でのデフォルト設定  
(-cpu=sh2 -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)

### 1.4 関連アプリケーションノート

- 「SH7619 グループ CPG 動作周波数変更時の設定例」
- 「SH7619 グループ BSC SDRAM インタフェース設定例（16 ビットバス）」
- 「SH7619 グループ BSC フラッシュメモリ接続例」
- 「SH7619 グループ キャッシュ設定例」

## 2. 応用例の説明

C 言語で作成されたメイン関数を実行するためには、パワーオンリセット後に、メモリの初期化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。本資料では初期設定プログラムにおける初期設定例を説明します。

SH7619 の各アプリケーションノートでは、本資料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、本体の PowerON\_Reset\_PC 関数を含む resetprg.c と、コールされる関数側の hwsetup.c や init\_section.c などの複数のソースファイルから構成されています。以下に主要なソースファイルを説明します。

- resetprg.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関数は、リセット解除後最初に行われる関数で、vecttbl.cで定義するリセットベクタにこの関数の先頭アドレスが設定されています。図1にPowerON\_Reset\_PC関数のフローを示します。
- hwsetup.c**  
 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関数は、クロックパルス発振器(CPG)、バスステートコントローラ(BSC)、キャッシュを設定する関数をコールして、システムに最低限必要なハードウェア設定を行います。図2にHardwareSetup関数の処理フローを示します。

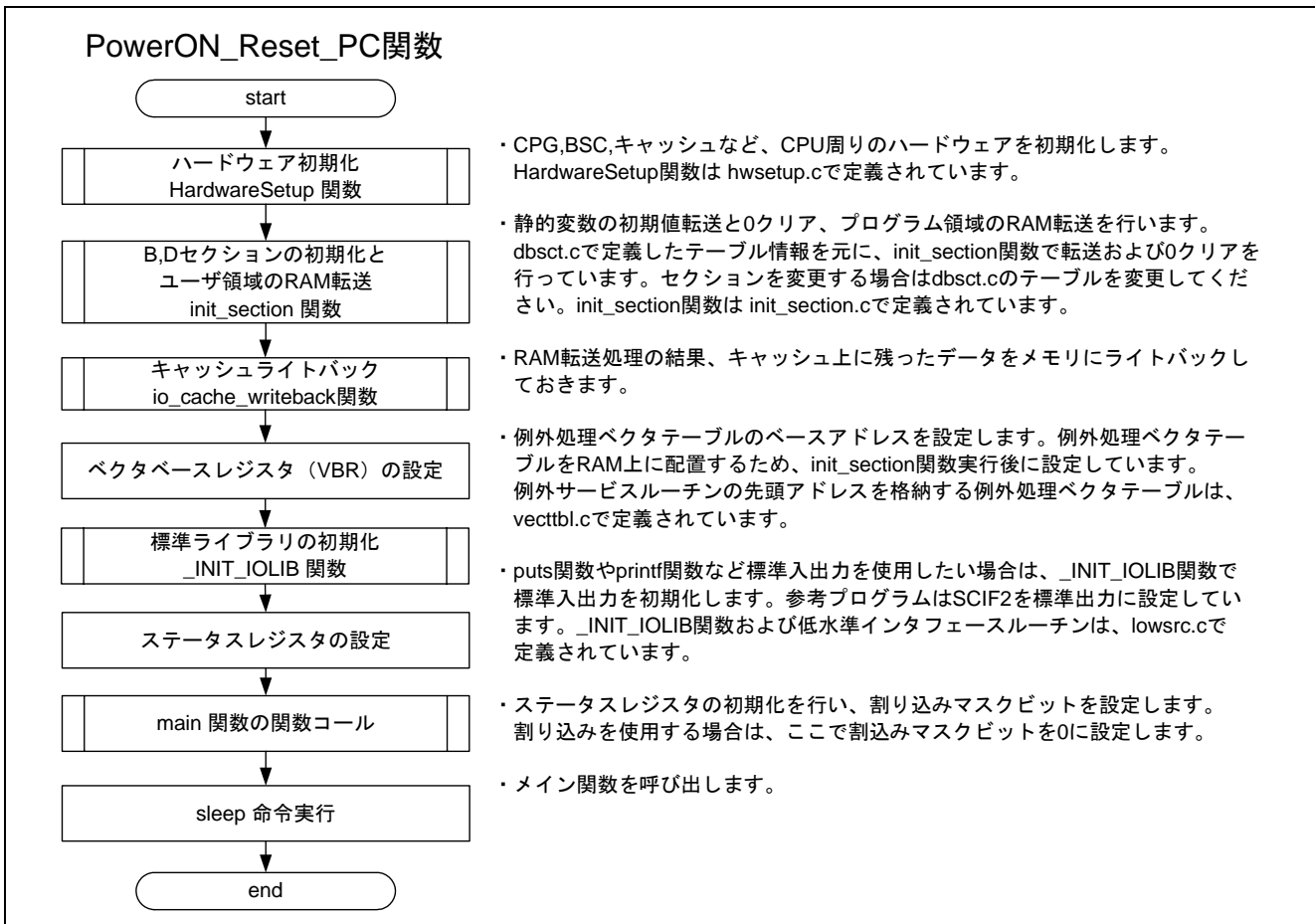


図1 リセットプログラムの処理フロー

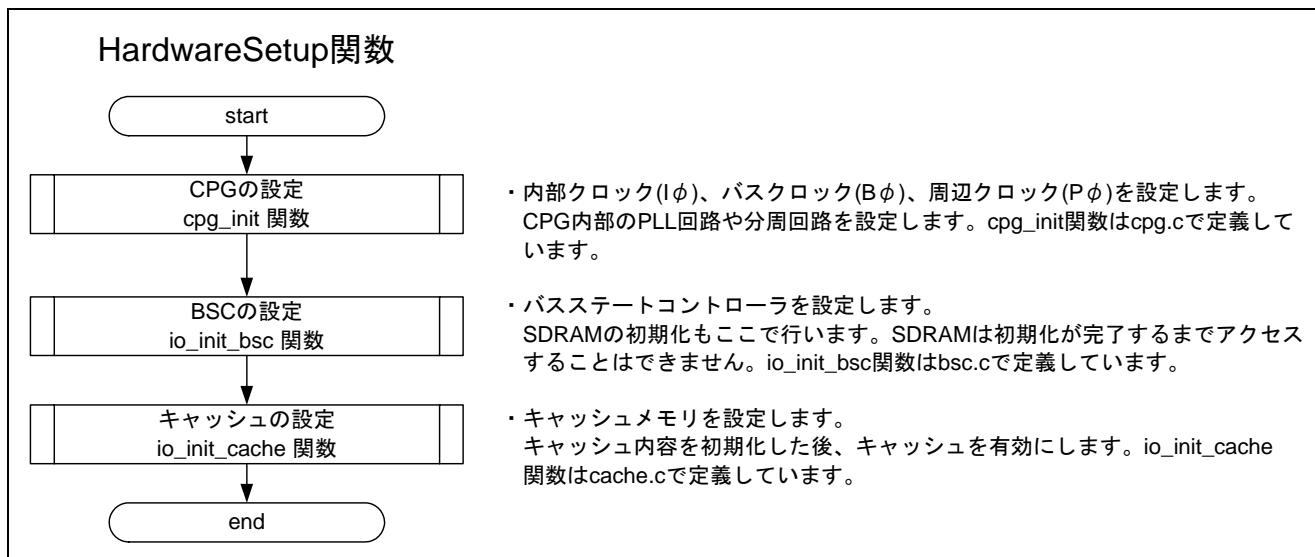


図2 ハードウェア初期化関数の処理フロー

## 2.2 参考プログラムにおける設定内容

表 1に参考プログラムでの設定を示します。

表1 参考プログラムでの設定

モジュール	設定内容
CPG	内部クロック : 125MHz バスクロック : 62.5MHz 周辺クロック : 31.25MHz
BSC	CS0 : フラッシュメモリ アクセスウェイトサイクル数 : 4 サイクル CS3 : SDRAM データバス幅 : 16 ビット ロウアドレスビット : 12 ビット カラムアドレスビット : 9 ビット CAS レイテンシ : 2 サイクル
PFC	CS0 および CS3 で使用するアドレスバス、データバス、およびバス制御端子を設定
キャッシュ	有効
SCIF	標準出力として設定 ・ チャンネル 2 ・ 非同期/データ 8 ビット/パリティなし/1 ストップビット ・ 57600bps

## 2.3 参考プログラム使用時の注意点

- SDRAM 初期化前に SDRAM をアクセスしないでください。  
 参考プログラムは、HardwareSetup 関数でバスステートコントローラを初期化した後に、SDRAM 上の領域を使用しています。未初期化の SDRAM を使用した場合は正常に動作しませんのでご注意ください。
- スタック領域である S セクションは SDRAM 上に配置しないでください。  
 スタックポインタ(R15)の初期値は、リセットベクタに設定されている値(S セクションの最終番地+1)が設定されます。参考プログラムは S セクションを内蔵メモリ上に配置するため問題ありませんが、SDRAM 上に配置すると初期設定プログラム内の関数コール実行時に、未初期化の SDRAM をアクセスしてしまいます。
- init\_section 関数実行前に静的変数領域をアクセスしないでください。  
 C 言語の静的変数領域は、init\_section 関数の実行によって初期化されます。それ以前にアクセスした場合は値が不定となりますのでご注意ください。

### 3. ユーザプログラム領域の RAM 転送設定

本サンプルプログラムでは、main 関数などのユーザ領域におけるプログラム領域と定数領域を、RAM に転送して実行しています。ここではその仕組みと設定内容を説明します。

#### 3.1 参考プログラムのセクション配置

参考プログラムのセクション配置を表 2 で示します。また SHC コンパイラのメモリ領域とセクション名の対応を表 3 で示します。

参考プログラムは、高速化のため、ROM 領域の P セクションを、RAM 領域の RP セクションに転送して実行します。P セクションにはメイン関数を含むユーザプログラム領域の他、標準ライブラリも含まれます。その他、C セクションと DINTTBL も RAM 上に転送して高速化しています。

初期設定プログラムなど RAM 上に転送できないものもあります。その場合は「#pragma section」機能を使用して PResetPRG セクションまたは PIntPRG セクションに配置しています。これらのセクションは転送対象から外しています。

コンパイラに関する詳細は「SuperH RISC engine C/C++コンパイラ、アセンブラ、最適化リンケージエディタコンパイラパッケージ V.9.01 ユーザーズマニュアル」の「4. 最適化リンケージエディタの操作方法」や「9.1 プログラムの構造」をご参照ください。

表2 参考プログラムのセクション配置

アドレス	デバイス	セクション	説明
0x00000000	FlashROM	DVECTTBL	リセットベクタ
		DINTTBL	例外処理ベクタテーブル
PResetPRG		初期設定プログラムのプログラム領域	
PIntPRG		NMI 割込みなどの割込みプログラム領域	
C\$BSEC		dbstc.c で定義した 0 クリア処理用のテーブル	
C\$DSEC		dbstc.c で定義した転送処理用のテーブル	
P		ユーザプログラムおよび標準ライブラリのプログラム領域	
C		ユーザプログラムおよび標準ライブラリの定数領域	
D		ユーザプログラムおよび標準ライブラリの初期化データ領域(初期値)	
0x0C000000		SDRAM	RP
	RC		C セクションの RAM 転送先
0xA0000800	FlashROM (キャッシュ無効)	PCACHE	非キャッシュ領域に配置するプログラム
0xE55FC000	内蔵 RAM	RINTTBL	例外処理ベクタテーブルの RAM 転送先
		R	ユーザプログラムおよび標準ライブラリの初期化データ領域(変数)
		B	ユーザプログラムおよび標準ライブラリの未初期化データ領域
0xE55FFBF0		S	スタック領域



表3 コンパイラが管理するメモリ領域とセクション

メモリ領域	セクション <sup>*1</sup>	内容
プログラム領域	P	機械語を格納
定数領域	C	const 型のデータを格納
初期化データ領域	D <sup>*2</sup>	初期値のあるデータを格納
未初期化データ領域	B	初期値のないデータを格納

\*1 「#pragma section」を使用すると別名のセクションを指定可能です。ただし、指定したセクション名の先頭には、上記のセクション名が添付されます。例) 「#pragma section ResetPRG」 → PResetPRG / CResetPRG / DResetPRG / BResetPRG

\*2 RAM 上に配置する変数領域は、最適化リンケージエディタのセクションオプションで定義します。通常 R セクションとします。同出力オプションの ROM 化支援機能において ROM : D、RAM : R と設定すると、シンボルが RAM 上のアドレスでリロケーション解決されます。

## 3.2 RAM 転送設定

ROM 領域のセクションを、RAM 領域のセクションに転送する手順を説明します。

### 3.2.1 dbstc.c の転送テーブル設定

dbstc.c は初期値転送用テーブル DTBL[ ]と、0 クリア用テーブル BTBL[ ]を定義しています。転送したいセクションを DTBL[ ]に設定します。

### 3.2.2 init\_section 関数

init\_section関数は「3.2.1 dbstc.cの転送テーブル設定」で設定したセクションを、転送または0クリアします。初期設定プログラムの中で、SDRAM初期化後、実行します。

なお、init\_section関数は標準ライブラリ\_INITSCT関数と同一の機能ですが、参考プログラムではPセクションを転送対象とするために、\_INITSCT関数を使用していません。\_INITSCT関数はデフォルトでPセクションに配置されるため、INITSCT関数自体が転送対象となり転送処理が行えません。参考プログラムではinit\_section関数をRAMに転送しないPRResetPRGセクションに配置していますので、ユーザプログラムや標準ライブラリを含むPセクションを転送対象とすることができます。

### 3.2.3 ROM 化支援機能

ROM 上のデータやプログラムを RAM 上に転送して実行する場合は、該当箇所のメモリ内容をコピーするだけでは不十分です。リンク時に ROM セクション内定義シンボルが RAM セクション内アドレスになるよう設定を行います。

最適化リンケージエディタの「出力カテゴリ」-「ROM 化支援」オプション (High-performance Embedded Workshop のビルドメニューから SuperH RISC Engine Standard Toolchain を開き、最適化リンカタブのカテゴリ : 出力、オプション項目 : ROM から RAM にマップするセクション を選択) で、転送したい ROM セクションと RAM セクションを設定します。このオプションにより、シンボル情報が RAM 上のアドレスでリロケーション解決されます。

## 4. 参考プログラムリスト

### 4.1 サンプルプログラムリスト"resetprg.c" (1)

```

1  /*"FILE COMMENT"*****
2  *
3  *   System Name : SH7619 Sample Program
4  *   File Name   : resetprg.c
5  *   Version    : 1.02.01
6  *   Contents   : SH7619 初期化処理
7  *   Model      : M3A-HS19
8  *   CPU        : SH7619
9  *   Compiler   : SHC9.1.1.0
10 *   OS         : none
11 *
12 *   note       : 本ファイルはRenesas Project Generator (Ver.4.06) により
13 *               生成されたファイルをアプリケーションノート用に変更しています。
14 *
15 *               <注意事項>
16 *               本サンプルプログラムはすべて参考資料であり、
17 *               その動作を保証するものではありません。
18 *               本サンプルプログラムはお客様のソフトウェア開発時の
19 *               技術参考資料としてご利用ください。
20 *
21 *   Copyright (C) 2007 Renesas Technology Corp. All Rights Reserved
22 *   AND Renesas Solutions Corp. All Rights Reserved
23 *
24 *   history    : 2007.06.29 ver.1.00.00
25 *               : 2007.07.09 ver.1.01.00 io_cache_writeback 追加
26 *               : 2007.09.28 ver.1.02.00 VBR 設定を RAM 転送後に変更
27 *               : 2007.11.19 ver.1.02.01 コメント一部修正
28 *"FILE COMMENT END"*****/
29 #include <stdio.h>
30 #include <stdlib.h>
31 #include <machine.h>
32 #include <_h_c_lib.h>
33 #include "stacksct.h"
34 #include "defs.h"
35 #include "iodefine.h"
36
37 #define SR_Init    0x00000000
38 #define INT_OFFSET 0x10
39
40 extern unsigned int INT_Vectors;
41 void PowerON_Reset_PC(void);
42 void main(void);
43 extern void HardwareSetup(void);
44 extern void init_section(void);
45 extern int io_cache_writeback(void);
46
47
48 #pragma section ResetPRG
49 #pragma entry PowerON_Reset_PC
    
```

## 4.2 サンプルプログラムリスト"resetprg.c" (2)

```

50  /*"FUNC COMMENT"*****
51  * ID      :
52  * Outline : CPU 初期化処理
53  *-----
54  * Include : #include "iodefine.h"
55  *-----
56  * Declaration : void PowerON_Reset_PC(void)
57  *-----
58  * Function  : パワーオンリセット例外ベクタテーブルに登録する CPU 初期化処理です。
59  *           : パワーオンリセット後本関数を最初に最初に実行します。
60  *-----
61  * Argument  : void
62  *-----
63  * ReturnValue : void
64  *-----
65  * Notice    :
66  *"FUNC COMMENT END"*****/
67 void PowerON_Reset_PC(void)
68 {
69     /* ==== Hardware setup ==== */
70     HardwareSetup();
71
72     /*==== B,Dセクションの初期化 ====*/
73     // _INITSCT();           /* 標準ライブラリを含め RAM に転送するため未使用 */
74     init_section();
75     io_cache_writeback();
76
77     /*==== ベクタベースレジスタ (VBR) の設定 ====*/
78     set_vbr((void *)((char *)&INT_Vectors - INT_OFFSET));
79
80     /* ==== 標準ライブラリの初期化 ==== */
81     _INIT_IOLIB();         // Use stdio I/O
82
83     //errno=0;              // Remove the comment when you use errno
84     //srand((_UINT)1);      // Remove the comment when you use rand()
85     //_slp_ptr=NULL;        // Remove the comment when you use strtok()
86
87     /*==== ステータスレジスタの設定 ====*/
88     set_cr(SR_Init);       /* 割込み許可 */
89     nop();
90
91     /*==== main 関数の関数コール ====*/
92     main();
93
94     /*==== sleep 命令実行 ====*/
95     sleep();
96 }
97
98 /* End of File */

```

### 4.3 サンプルプログラムリスト"hwsetup.c" (1)

```

1  /*"FILE COMMENT"*****
2  *
3  *   System Name : SH7619 Sample Program
4  *   File Name   : hwsetup.c
5  *   Version    : 1.00.00
6  *   Contents   : ハードウェア初期化関数
7  *   Model     : M3A-HS19
8  *   CPU       : SH7619
9  *   Compiler  : SHC9.1.1.0
10 *   OS        : none
11 *
12 *   note      : 本ファイルはRenesas Project Generator (Ver.4.06) により
13 *              生成されたファイルをアプリケーションノート用に変更しています。
14 *
15 *              <注意事項>
16 *              本サンプルプログラムはすべて参考資料であり、
17 *              その動作を保証するものではありません。
18 *              本サンプルプログラムはお客様のソフトウェア開発時の
19 *              技術参考資料としてご利用ください。
20 *
21 *   Copyright (C) 2007 Renesas Technology Corp. All Rights Reserved
22 *   AND Renesas Solutions Corp. All Rights Reserved
23 *
24 *   history   : 2007.06.29 ver.1.00.00
25 *"FILE COMMENT END"*****/
26 #include "defs.h"
27 #include "iodefine.h"
28
29 /* ==== Prototype declaration ==== */
30 void HardwareSetup(void);
31
32 /* ==== referenced external Prototype declaration ==== */
33 extern void io_set_cpg(void);
34 extern void io_init_bsc(void);
35 extern void io_init_cache(void);
36
37 #pragma section ResetPRG
38 /*"FUNC COMMENT"*****
39 * ID      :
40 * Outline : Hardware initialization function
41 *-----
42 * Include : #include "iodefine.h"
43 *-----
44 * Declaration : void HardwareSetup(void)
45 *-----
46 * Function   : SH7619 のハードウェア初期化
47 *-----
48 * Argument   : void
49 *-----
50 * ReturnValue : void
51 *-----
52 * Notice    :
53 *"FUNC COMMENT END"*****/

```

#### 4.4 サンプルプログラムリスト"hwsetup.c" (2)

```

54 void HardwareSetup(void)
55 {
56     /*==== CPG setting ====*/
57     cpg_init();
58
59     /*==== BSC initialize( CS0 : フラッシュ、 CS3 : SDRAM ) ====*/
60     io_init_bsc();
61
62     /*==== Cache setting ====*/
63     io_init_cache();
64 }
65
66 /* End of File */
    
```

#### 4.5 サンプルプログラムリスト"cpg\_init.c" (1)

```

1  /*"FILE COMMENT"*****
2  *
3  *      System Name : SH7619 Sample Program
4  *      File Name   : cpg_init.c
5  *      Version    : 1.00.00
6  *      Contents   : CPG 設定処理
7  *      Model     : M3A-HS19
8  *      CPU       : SH7619
9  *      Compiler  : SHC9.1.1.0
10 *      OS        : none
11 *
12 *      note      : 本ファイルはRenesas Project Generator (Ver.4.06) により
13 *                生成されたファイルをアプリケーションノート用に変更しています。
14 *
15 *                <注意事項>
16 *                本サンプルプログラムはすべて参考資料であり、
17 *                その動作を保証するものではありません。
18 *                本サンプルプログラムはお客様のソフトウェア開発時の
19 *                技術参考資料としてご利用ください。
20 *
21 *      Copyright (C) 2007 Renesas Technology Corp. All Rights Reserved
22 *      AND Renesas Solutions Corp. All Rights Reserved
23 *
24 *      history   : 2007.06.29 ver.1.00.00
25 *"FILE COMMENT END"*****
26 #include "defs.h"
27 #include "iodefine.h"
28
29 /* ==== プロトタイプ宣言 ==== */
30 void cpg_init(void);
31
32 #pragma section ResetPRG
    
```

## 4.6 サンプルプログラムリスト"cpg\_init.c" (2)

```

33  /*"FUNC COMMENT"*****
34  * ID      :
35  * Outline : CPG の設定
36  *-----
37  * Include : #include "iopdefine.h"
38  *-----
39  * Declaration : void cpg_init(void)
40  *-----
41  * Function  : クロックパルス発振器 (CPG)の設定を行い、内部クロック
42  *            : (I Clock)、周辺クロック (P Clock)、バスクロック
43  *            : (B Clock) を (I:B:P)=(125MHz:62.5MHz:31.25MHz)に設定
44  *            : します。
45  *-----
46  * Argument  : void
47  *-----
48  * Returnvalue : void
49  *-----
50  * Notice    : 本機能は、入力クロック : 15.625MHz、クロックモード : 1 の
51  *            : 場合の設定例です。
52  *"FUNC COMMENT END"*****/
53  void cpg_init(void)
54  {
55      /* ==== CPG 通倍率と分周率の変更 ==== */
56      /* ---- WDT 設定 ---- */
57      WDT.WRITE.WTCSR = 0xa507;      /* 停止+クロックセレクト Pφ/4096*/
58                                      /* 上位 : 0xA5、ワードアクセス */
59      WDT.WRITE.WTCNT = 0x5aB2;     /* 周波数安定時間 10ms を確保 */
60                                      /* (0xFF-0xB2)*4096/31.25MHz>10ms*/
61                                      /* 上位 : 0x5A、ワードアクセス */
62      /* ---- モード端子制御の設定 ---- */
63      SBY.CR.BIT.MDCHG = 0;        /* ソフトウェアスタンバイによるモード *
64                                      * 変更が有効の場合、FRQCR の設定はソフ*
65                                      * トウェアスタンバイ解除後まで反映さ *
66                                      * れない。 */
67      /* ---- CPG 設定 ---- */
68      CPG.FRQCR.WORD = 0x1103;     /* Iφ = 125MHz */
69                                      /* Bφ = 62.5MHz */
70                                      /* Pφ = 31.25MHz((PLL×2×4)/4) */
71                                      /* CPU mode1,PLL1×2,PLL2×4 */
72  }
73
74  /* End of File */
    
```

#### 4.7 サンプルプログラムリスト"bsc.c" (1)

```

1  /*"FILE COMMENT"*****
2  *
3  *      System Name : SH7619 Sample Program
4  *      File Name   : bsc.c
5  *      Version    : 1.01.00
6  *      Contents   : SH7619 初期設定
7  *      Model     : M3A-HS19-B
8  *      CPU       : SH7619
9  *      Compiler  : SHC9.1.1.0
10 *      OS        : none
11 *
12 *      note      : 本ファイルはRenesas Project Generator (Ver.4.06) により
13 *                生成されたファイルをアプリケーションノート用に変更しています。
14 *
15 *                <注意事項>
16 *                本サンプルプログラムはすべて参考資料であり、
17 *                その動作を保証するものではありません。
18 *                本サンプルプログラムはお客様のソフトウェア開発時の
19 *                技術参考資料としてご利用ください。
20 *
21 *      Copyright (C) 2008 Renesas Technology Corp. All Rights Reserved
22 *      AND Renesas Solutions Corp. All Rights Reserved
23 *
24 *      history   : 2007.06.29 ver.1.00.00
25 *                : 2008.01.09 ver.1.01.00 flash memory(renesas -> spansion)
26 *"FILE COMMENT END"*****/
27 #include "defs.h"
28 #include "iodefine.h"
29
30 /* ==== プロトタイプ宣言 ==== */
31 void io_init_bsc(void);
32
33 #pragma section ResetPRG
34 /*"FUNC COMMENT"*****
35 * ID          :
36 * Outline     : BSC の設定
37 *-----
38 * Include     : #include "iodefine.h"
39 *-----
40 * Declaration : void io_init_bsc(void)
41 *-----
42 * Function    : ピンファンクションコントローラ (PFC)およびバスステート
43 *              : コントローラ (BSC) の設定を行い、CS0 空間のフラッシュメモリと
44 *              : CS3 空間の SDRAM に対するアクセスタイミングを設定します。
45 *-----
46 * Argument    : void
47 *-----
48 * ReturnValue : void
49 *-----
50 * Notice      : PFC の設定は他の処理で設定した PFC 設定値を変更しないように
51 *              : ビット操作により設定しています。
52 *"FUNC COMMENT END"*****/

```

## 4.8 サンプルプログラムリスト"bsc.c" (2)

```

53 void io_init_bsc( void )
54 {
55     volatile int j = LOOP_100us * 2;
56
57     /* ==== PFC の設定 ==== */
58     /* ---- 共通 ----*/
59     PFC.PACRH2.BIT.PA21MD = 1;      /* Set A21-A16 */
60     PFC.PACRH2.BIT.PA20MD = 1;      /* リセット後、アドレスバス[A16-A25]は */
61     PFC.PACRH2.BIT.PA19MD = 1;      /* 入力ポートのため、設定が必要。 */
62     PFC.PACRH2.BIT.PA18MD = 1;      /* ブート時に上記端子が必要な場合は、 */
63     PFC.PACRH2.BIT.PA17MD = 1;      /* プルダウン/プルアップが必要 */
64     PFC.PACRH2.BIT.PA16MD = 1;
65     /* ---- SDRAM 向け ---- */
66     PFC.PBCRL1.BIT.PB12MD = 1;      /* Set CS3# */
67     PFC.PBCRL2.BIT.PB2MD = 1;      /* Set CKE */
68     PFC.PBCRL2.BIT.PB3MD = 1;      /* Set CASL */
69     PFC.PBCRL2.BIT.PB4MD = 1;      /* Set RASL */
70
71
72     /* ==== CS0 空間 (フラッシュメモリ) ==== */
73     BSC.CS0BCR.LONG = 0x00000400; /* WR-RD/WR-WR サイクル間アイドル = 0 *
74                                     * 別空間 RD-WR サイクル間アイドル= 0 *
75                                     * 同空間 RD-WR サイクル間アイドル= 0 *
76                                     * 別空間 RD-RD サイクル間アイドル= 0 *
77                                     * 同空間 RD-RD サイクル間アイドル= 0 *
78                                     * メモリ種別 = 通常空間 *
79                                     * データバス幅 = 16 ビット */
80
81     BSC.CS0WCR = 0x00000ac0; /* Addr/CS→RD/WE 遅延サイクル = 1.5 *
82                               * アクセスウェイトサイクル = 5 *
83                               * 外部ウェイト入力無視 *
84                               * RD/WE→アドレス/CS 遅延サイクル= 0.5 */
85
86
87     /* ==== CS3 空間 (SDRAM) ==== */
88     BSC.CS3BCR.LONG = 0x00004400; /* WR-RD/WR-WR サイクル間アイドル = 0 *
89                                     * 別空間 RD-WR サイクル間アイドル= 0 *
90                                     * 同空間 RD-WR サイクル間アイドル= 0 *
91                                     * 別空間 RD-RD サイクル間アイドル= 0 *
92                                     * 同空間 RD-RD サイクル間アイドル= 0 *
93                                     * メモリ種別 = SDRAM *
94                                     * データバス幅 = 16 ビット */
95
96     BSC.CS3WCR = 0x00002892; /* プリチャージ完了待ちウェイト =1 *
97                               * ACTV→READ/WRITE コマンド間ウェイト=2 *
98                               * CAS レイテンシ =2 *
99                               * プリチャージ起動待ちウェイト =2 *
100                              * REF/セルフリフレッシュ解除→ *
101                              * ACTV/REF/MRS コマンド間アイドル =5 */
102

```



## 4.9 サンプルプログラムリスト"bsc.c" (3)

```

103     BSC.SDCR.LONG = 0x00000809; /* オートリフレッシュを行う          *
104                               * オートプリチャージモード          *
105                               * エリア 3 ロウアドレス = 12 ビット      *
106                               * エリア 3 カラムアドレス = 9 ビット    */
107
108     BSC.RTCR.LONG = 0xa55a0010; /* << 0xa55axxxx : ライトプロテクト >> *
109                               * クロックセレクト = Bφ/16 (256nsec)    *
110                               * 連続リフレッシュ回数 = 1          */
111
112     BSC.RTCNT      = 0xa55a0000; /* << 0xa55axxxx : ライトプロテクト >> *
113                               * カウンタクリア          */
114
115     BSC.RTCOR      = 0xa55a003c; /* << 0xa55axxxx : ライトプロテクト >> *
116                               * 平均リフレッシュ間隔[15.6us]      *
117                               * 15600/256 = 61(60(0x3c)+1)        */
118
119     /* ---- SDRAM モードレジスタへの書き込み ---- */
120     while(j-- > 0){
121         /* 200μsec wait */
122     }
123     *(volatile unsigned short *)0xf8fd5040 = 0;
124                               /* アドレスで設定 (ダミーワードライト) *
125                               * バーストリード/バーストライト      *
126                               * バースト長1、データバス幅 16 ビット *
127                               * CAS レイテンシ 2          */
128 }
129
130 /* End of File */
    
```

## 4.10 サンプルプログラムリスト"cache.c" (1)

```

1  /*"FILE COMMENT"*****
2  *
3  *      System Name : SH7619 Sample Program
4  *      File Name   : cache.c
5  *      Version     : 1.01.00
6  *      Contents    : sample of cache register
7  *      Model       : M3A-HS19
8  *      CPU         : SH7619
9  *      Compiler    : SHC9.1.1.0
10 *      OS          : none
11 *
12 *      note        : It is a reference program for cache setting.
13 *
14 *      <注意事項>
15 *      本サンプルプログラムはすべて参考資料であり、
16 *      その動作を保証するものではありません。
17 *      本サンプルプログラムはお客様のソフトウェア開発時の
18 *      技術参考資料としてご利用ください。
19 *
20 *
21 *      The information described here may contain technical inaccuracies or
22 *      typographical errors. Renesas Technology Corporation and Renesas Solutions
23 *      assume no responsibility for any damage, liability, or other loss rising
24 *      from these inaccuracies or errors.
25 *
26 *      Copyright (C) 2007 Renesas Technology Corp. All Rights Reserved
27 *      AND Renesas Solutions Corp. All Rights Reserved
28 *
29 *      history     : 2007.06.29 ver.1.00.00
30 *                  : 2007.07.09 ver.1.01.00 *io_cache_writeback 追加
31 *"FILE COMMENT END"*****/
32 #include <machine.h>
33 #include "defs.h"
34 #include "iodefine.h"
35
36 /* ==== Prototype Declaration ==== */
37 void io_init_cache(void);
38
39 #pragma section CACHE      /* このセクションは CS0 領域の非キャッシュ領域に配置 */
    
```

## 4.11 サンプルプログラムリスト"cache.c" (2)

```

40  /*"FUNC COMMENT"*****
41  * ID      :
42  * Outline : cacheの初期化
43  *-----
44  * Include : #include "iodefine.h"
45  *-----
46  * Declaration : void io_init_cache(void)
47  *-----
48  * Function  : キャッシュをフラッシュした後、キャッシュを有効にします。
49  *          :
50  *-----
51  * Argument  : void
52  *-----
53  * Returnvalue : void
54  *-----
55  * Notice    : この関数はCS0領域の非キャッシュ領域に配置してください。
56  *          : キャッシュ操作中は割り込みを禁止しています。
57  *"FUNC COMMENT END"*****/
58  void io_init_cache(void)
59  {
60      volatile unsigned long reg;
61      int mask;
62
63      /* ==== 割り込み禁止設定 ==== */
64      mask = get_imask();
65      set_imask(15);
66
67      /* ==== キャッシュ制御レジスタ1 (CCR1) の設定 ==== */
68      CCR1.LONG = 0x0000009ul;      /*
69                                     CF=1:Cache flash
70                                     CE=1:Instruction cache enabled
71                                     */
72
73      /* ==== 割り込みマスクレベルを戻す ==== */
74      set_imask(mask);
75
76  }
77
    
```

## 4.12 サンプルプログラムリスト"cache.c" (3)

```

78  /*"FUNC COMMENT"*****
79  * ID      :
80  * Outline  : キャッシュのライトバック
81  *-----
82  * Include  : #include "iodefine.h"
83  *-----
84  * Declaration: int io_cache_writeback(void);
85  *-----
86  * Function  : キャッシュ上の変更されたデータを、全てメモリに書き戻します。
87  *-----
88  * Argument  : None
89  *-----
90  * ReturnValue: 0 : 正常終了
91  *-----
92  * Notice    :
93  *"FUNC COMMENT END"*****/
94  int io_cache_writeback(void)
95  {
96      volatile unsigned long *array;
97      unsigned int i,j;
98      int mask;
99
100     mask = get_imask();          /* ↓↓↓ 割込み禁止 ↓↓↓ */
101     set_imask(15);
102
103     /* ==== キャッシュエントリをメモリにライトバック ==== */
104     for(i=0u; i <4u; i++){      /* 4 ウェイ */
105         for(j=0u; j < 256u; j++){ /* 128 エントリ */
106             ;
107             /* ---- 各アドレスアレイへ書き込むと、メモリに書き戻される ---- */
108             /* アドレスを計算:          0xf0000000 + way + entry + A */
109             array = (volatile unsigned long *) (0xf0000000 | (i<<12ul) | (j<<4ul) | 0x8 );
110             /* データを書き込む(U=V=0, 他はそのまま)*/
111             *array &= 0xffffffffcul; /* U=V=1 のエントリが書き戻される */
112         }
113     }
114     set_imask(mask);          /* ↑↑↑ 割込み禁止 ↑↑↑ */
115
116     return 0;
117 }
118
119 /* End of File */

```

## 5. 参考ドキュメント

- ソフトウェアマニュアル  
SH-1/SH-2/ SH-DSP ソフトウェアマニュアル Rev7.00  
(最新版をルネサス テクノロジホームページから入手してください)。
- ハードウェアマニュアル  
SH7619 グループハードウェアマニュアル Rev.5.00  
(最新版をルネサス テクノロジホームページから入手してください)。

## ホームページとサポート窓口

ルネサステクノロジホームページ

<http://japan.renesas.com/>

お問い合わせ先

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

[csc@renesas.com](mailto:csc@renesas.com)

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2007.12.28	—	第 1 版作成

### 本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりますは、事前に弊社営業窓口で最新の情報をご確認頂きますとともに、弊社ホームページ (<http://www.renesas.com>) などを通じて公開される情報に常にご注意下さい。
5. 本資料に記載した情報は、正確を期すため慎重に制作したのですが、万一本資料の記述の誤りに起因する損害がお客様に生じた場合においても、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断して下さい。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会下さい。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないで下さい。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
  - 1) 生命維持装置。
  - 2) 人体に埋め込み使用するもの。
  - 3) 治療行為（患部切り出し、薬剤投与等）を行なうもの。
  - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて製品をご使用された場合の故障および事故につきましては、弊社はその責任を負いません。
10. 弊社は製品の品質及および信頼性の向上に努めておりますが、特に半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。弊社製品の故障または誤動作が生じた場合も人身事故、火災事故、社会的損害などを生じさせないよう、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特にマイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願い致します。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断り致します。
13. 本資料に関する詳細についてのお問い合わせ、その他お気付きの点等がございましたら弊社営業窓口までご照会下さい。