
SH7785 グループ

R01AN0242JJ0102

Rev.1.02

SH7785 初期設定例

2012.04.02

要旨

この資料は、SH7785 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

SH7785

目次

| | |
|----------------------------------|----|
| 1. はじめに..... | 2 |
| 2. ローカルバスステートコントローラ(LBSC)..... | 4 |
| 3. DDR2-SDRAMインタフェース(DBSC2)..... | 8 |
| 4. 応用例の説明..... | 11 |
| 5. 参考プログラム例..... | 22 |
| 6. 参考ドキュメント..... | 76 |

1. はじめに

1.1 仕様

パワーオンリセット解除後に、ローカルバスステートコントローラ(LBSC)、DDR2-SDRAM インタフェース(DBSC2)の初期設定を行います。

1.2 使用機能

- ローカルバスステートコントローラ
- DDR2-SDRAM インタフェース(DBSC2)

1.3 適用条件

| | |
|-------------|--|
| 評価ボード: | ルネサスエレクトロニクス(株)製 R0P7785LC0011RL 外付けメモリ (エリア 0): NOR 型 Flash メモリ 64M バイト Spansion 製 S29GL256P90TFIRI (エリア 2, 3): DDR2-SDRAM 128M バイト (32ibt モード時 : 512M バイト) Elpida 製 EDE1108ACSE-6E-E (4 個) |
| マイコン: | SH7785 |
| 動作周波数: | 内部クロック 600MHz SuperHyway クロック 300MHz 周辺クロック 50MHz DDR2 クロック 300MHz 外部バスクロック 100MHz PCI バスクロック 33MHz |
| エリア 0 バス幅: | 32bit(MD5 端子=High レベル, MD6 端子=High レベル) |
| クロック動作モード: | クロックモード 16(MD0 端子=Low レベル, MD1 端子=Low レベル, MD2 端子=Low レベル, MD3 端子=Low レベル, MD4 端子=High レベル) |
| エンディアン: | リトルエンディアン(MD8 端子=High レベル) |
| アドレスモード: | 29 ビットアドレスモード(MD13 端子=Low レベル) |
| ツールチェーン: | Super-H RISC engine Standard Toolchain Ver9.3.2.0 |
| コンパイルオプション: | High-performance Embedded Workshop で include 指定以外はデフォルト設定 -cpu=sh4a -endian=little -include="\$ (PROJDIR)¥inc¥drv", "\$ (PROJDIR)¥inc" -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 |
| アセンブラオプション: | -cpu=sh4a -endian=little -round=zero -denormalize=off -include="\$ (PROJDIR)¥inc" -include="\$ (PROJDIR)¥inc¥drv" -debug -object="\$ (CONFIGDIR)¥\$(FILELEAF).obj" -literal=pool,branch,jump,return -nolist -nologo -chgincpath -errorpath |

表 1 に本応用例でのセクション配置を示します。

表 1 セクション配置

| セクション名 | セクション用途 | 領域 | 配置アドレス(仮想アドレス) | |
|------------|---------------------------|-----|----------------|---------------------------------|
| P | プログラム領域 | ROM | 0x00002000 | P0領域 (キャッシング可能, MMUアドレス変換可能) |
| C | 定数領域 | ROM | | |
| C\$BSEC | 未初期化データ領域用アドレス構造 | ROM | | |
| C\$DSEC | 初期化データ領域用アドレス構造 | ROM | | |
| D | 初期化データ | ROM | | |
| B | 未初期化データ領域 | RAM | 0x0C000000 | |
| R | 初期化データ領域 | RAM | | |
| S | スタック領域 | RAM | | |
| INTHandler | 例外/割込みハンドラ | ROM | 0x80001000 | P1領域 (キャッシング可能, MMUアドレス変換不可) |
| VECTTBL | リセットベクタテーブル 割込みベクタテーブル | ROM | | |
| INTTBL | 割込みマスクテーブル | ROM | | |
| PIntPRG | 割込み関数 | ROM | | |
| RSTHandler | リセットハンドラ | ROM | 0xA0000000 | P2領域 (キャッシング不可, MMUアドレス変換不可) |
| PResetPRG | リセットプログラム | ROM | | |
| PnonCACHE | プログラム領域 (キャッシュ無効アクセス) | ROM | | |

2. ローカルバスステートコントローラ(LBSC)

ローカルバスステートコントローラ (LBSC) は、外部メモリ空間の分割(チップ選択信号の設定)、各種メモリおよび接続デバイス仕様に応じた制御信号の出力などを行います。

2.1 レジスタ構成

表 2 にローカルバスステートコントローラ(LBSC)のレジスタ構成を示します。

表 2 LBSC レジスタ構成

| 名称 | 略称 | R/W | P4アドレス | アクセス サイズ | 同期 クロック |
|---------------------|--------|-----|-------------|-------------|------------|
| メモリアドレスマップ選択レジスタ | MMSELR | R/W | H'FC40 0020 | 32 | SHck |
| バスコントロールレジスタ | BCR | R/W | H'FF80 1000 | 32 | Bck |
| CS0バスコントロールレジスタ | CS0BCR | R/W | H'FF80 2000 | 32 | Bck |
| CS1バスコントロールレジスタ | CS1BCR | R/W | H'FF80 2010 | 32 | Bck |
| CS2バスコントロールレジスタ | CS2BCR | R/W | H'FF80 2020 | 32 | Bck |
| CS3バスコントロールレジスタ | CS3BCR | R/W | H'FF80 2030 | 32 | Bck |
| CS4バスコントロールレジスタ | CS4BCR | R/W | H'FF80 2040 | 32 | Bck |
| CS5バスコントロールレジスタ | CS5BCR | R/W | H'FF80 2050 | 32 | Bck |
| CS6バスコントロールレジスタ | CS6BCR | R/W | H'FF80 2060 | 32 | Bck |
| CS0ウェイトコントロールレジスタ | CS0WCR | R/W | H'FF80 2008 | 32 | Bck |
| CS1ウェイトコントロールレジスタ | CS1WCR | R/W | H'FF80 2018 | 32 | Bck |
| CS2ウェイトコントロールレジスタ | CS2WCR | R/W | H'FF80 2028 | 32 | Bck |
| CS3ウェイトコントロールレジスタ | CS3WCR | R/W | H'FF80 2038 | 32 | Bck |
| CS4ウェイトコントロールレジスタ | CS4WCR | R/W | H'FF80 2048 | 32 | Bck |
| CS5ウェイトコントロールレジスタ | CS5WCR | R/W | H'FF80 2058 | 32 | Bck |
| CS6ウェイトコントロールレジスタ | CS6WCR | R/W | H'FF80 1068 | 32 | Bck |
| CS5PCMCIAコントロールレジスタ | CS5PCR | R/W | H'FF80 1070 | 32 | Bck |
| CS6PCMCIAコントロールレジスタ | CS6PCR | R/W | H'FF80 1080 | 32 | Bck |

2.2 CnBCRの説明

CnBCR はアイドルサイクル、バス幅、メモリの種類を設定します。

図 1 に同一空間アイドルサイクル、図 2 に別空間アイドルサイクルのタイミングチャートを示します。

表 3 にタイミングのシンボルの説明を示します。

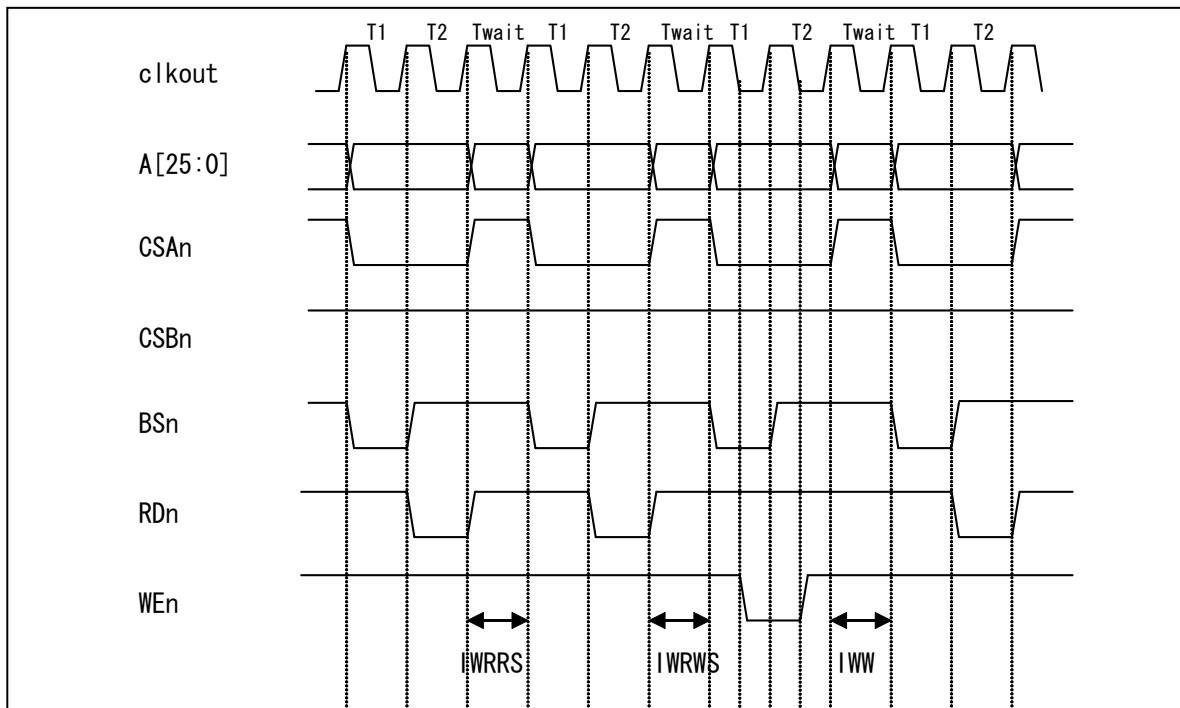


図 1 同一空間アイドルサイクル

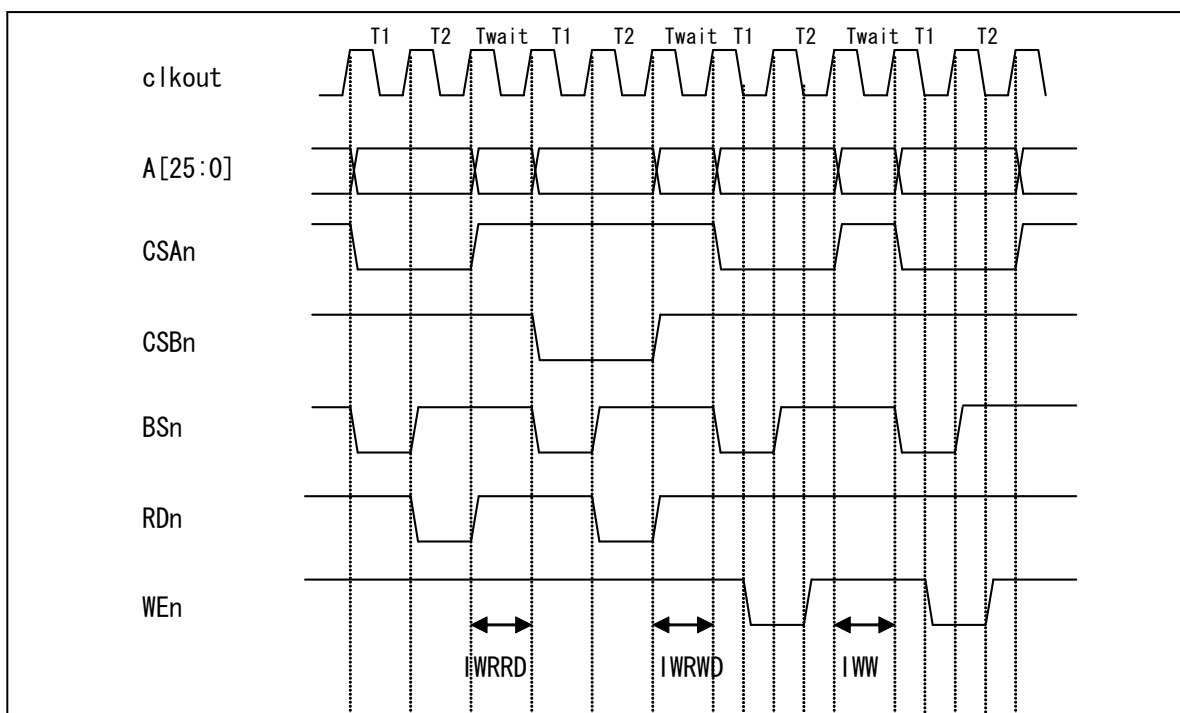


図 2 別空間アイドルサイクル

表 3 アイドルサイクルシンボル説明

| シンボル名 | 内容 |
|-------|--------------------------------------|
| IWRRS | 同一空間リード-リードサイクル間アイドル |
| IWRWS | 同一空間リード-ライトサイクル間アイドル |
| IWRRD | 別空間リード-リードサイクル間アイドル |
| IWRWD | 別空間リード-ライトサイクル間アイドル |
| IWW | ライト-リードサイクル間アイドル ライト-ライトサイクル間アイドル |

2.3 CnWCRの説明

CnWCR はアクセスサイクルのセットアップ/ホールド、ウェイトサイクルを設定します。

ライト時のデータの出力タイミングは ADS の設定により変わります。

図 3 にアクセスサイクルのタイミングチャートを示します。

表 4 にアクセスサイクルのシンボルの説明を示します。

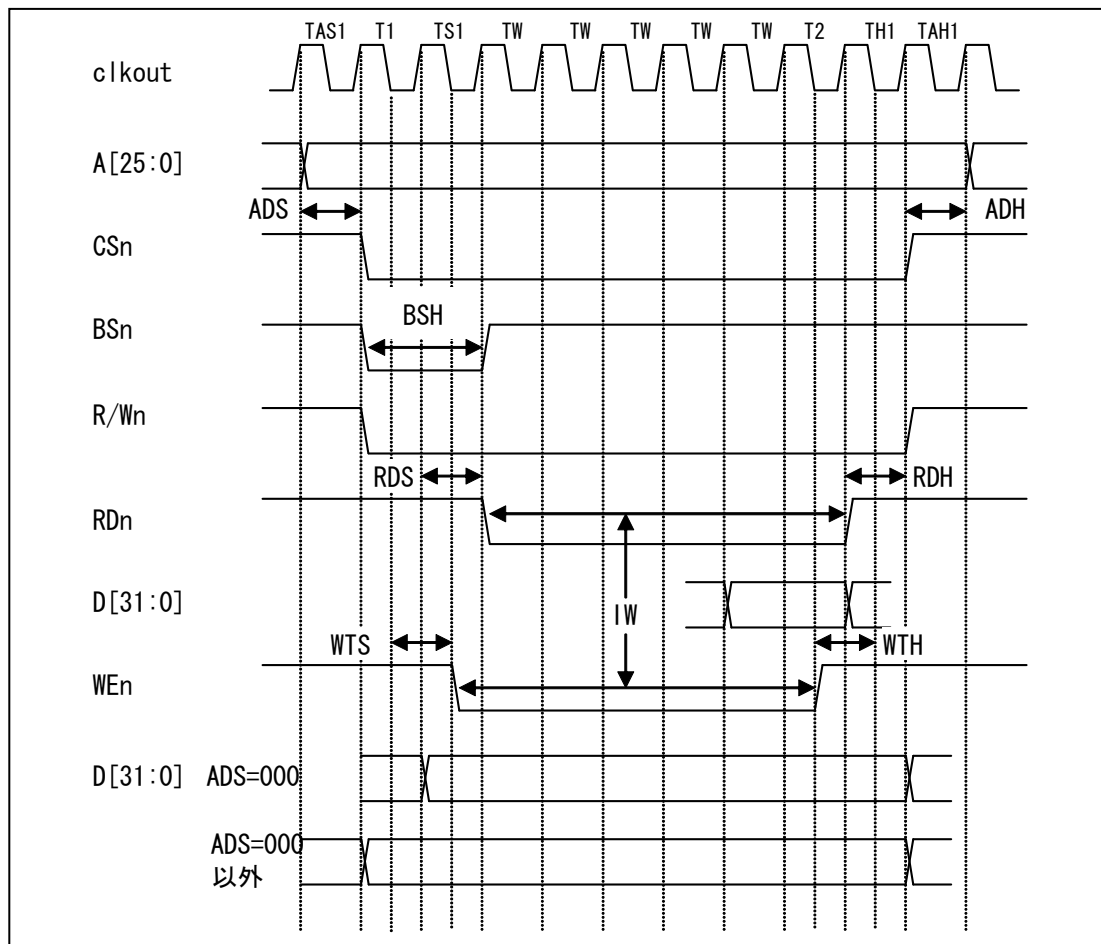


図 3 アクセスサイクルのタイミングチャート

表 4 アクセスサイクルのシンボル説明

| シンボル名 | 説明 |
|-------|----------------------|
| ADS | アドレスアサート-CSアサートの遅延期間 |
| ADH | CSネゲート-アドレスネゲートの遅延期間 |
| BSH | BSのサイクル数 |
| RDS | CSアサート-リードアサートの遅延期間 |
| RDH | リードネゲート-CSネゲートの遅延期間 |
| WTS | CSアサート-ライトアサートの遅延期間 |
| WTH | ライトネゲート-CSネゲートの遅延期間 |
| IW | リード及びライト期間のウェイトサイクル |

3. DDR2-SDRAMインタフェース(DBSC2)

DDR2-SDRAM インタフェース (DBSC2) は、CPU や各種モジュールからのアクセスを調停し、DDR2-SDRAM に対して制御信号を出力することにより、DDR2-SDRAM を接続することが可能です。

- 外部データバス幅は 16 ビット、32 ビットをサポート。
- DDR2-600(300MHz)と DDR2-400(200MHz)をサポート。
- バースト長 4 をサポート。
- 接続可能なメモリ容量、構成を表 5 に示します。

表 5 接続可能なメモリ容量と構成

| 外部データバス幅 | メモリ容量 | 構成 | 合計容量 |
|----------|--------------------|--------|---------|
| 16ビット | 256Mビット(16Mx16ビット) | 1個並列接続 | 256Mビット |
| | 256Mビット(32Mx8ビット) | 2個並列接続 | 512Mビット |
| | 512Mビット(32Mx16ビット) | 1個並列接続 | 512Mビット |
| | 512Mビット(64Mx8ビット) | 2個並列接続 | 1Gビット |
| | 1Gビット(64Mx16ビット) | 1個並列接続 | 1Gビット |
| | 1Gビット(128Mx8ビット) | 2個並列接続 | 2Gビット |
| | 2Gビット(128Mx16ビット) | 1個並列接続 | 2Gビット |
| | 2Gビット(256Mx8ビット) | 2個並列接続 | 4Gビット |
| 32ビット | 256Mビット(16Mx16ビット) | 2個並列接続 | 512Mビット |
| | 256Mビット(32Mx8ビット) | 4個並列接続 | 1Gビット |
| | 512Mビット(32Mx16ビット) | 2個並列接続 | 1Gビット |
| | 512Mビット(64Mx8ビット) | 4個並列接続 | 2Gビット |
| | 1Gビット(64Mx16ビット) | 2個並列接続 | 2Gビット |
| | 1Gビット(128Mx8ビット) | 4個並列接続 | 4Gビット |
| | 2Gビット(128Mx16ビット) | 2個並列接続 | 4Gビット |
| | 2Gビット(256Mx16ビット) | 4個並列接続 | 8Gビット |

3.1 レジスタ構成

表 6 に DBSC2 のレジスタ構成を示します。

表 6 DBSC2 レジスタ構成

| 名称 | 略称 | R/W | P4アドレス | アクセスサイズ | 同期クロック |
|---------------------------------|-------------|-----|-------------|---------|--------|
| DBSC2ステータスレジスタ | DBSTATE | R | H'FE80 000C | 32 | DDRck |
| SDRAM動作許可レジスタ | DBEN | R/W | H'FE80 0010 | 32 | DDRck |
| SDRAMコマンド制御レジスタ | DBCMDCNT | R/W | H'FE80 0014 | 32 | DDRck |
| SDRAM構成設定レジスタ | DBCONF | R/W | H'FE80 0020 | 32 | DDRck |
| SDRAMタイミングレジスタ0 | DBTR0 | R/W | H'FE80 0030 | 32 | DDRck |
| SDRAMタイミングレジスタ1 | DBTR1 | R/W | H'FE80 0034 | 32 | DDRck |
| SDRAMタイミングレジスタ2 | DBTR2 | R/W | H'FE80 0038 | 32 | DDRck |
| SDRAMリフレッシュ制御レジスタ0 | DBRFCNT0 | R/W | H'FE80 0040 | 32 | DDRck |
| SDRAMリフレッシュ制御レジスタ1 | DBRFCNT1 | R/W | H'FE80 0044 | 32 | DDRck |
| SDRAMリフレッシュ制御レジスタ2 | DBRFCNT2 | R/W | H'FE80 0048 | 32 | DDRck |
| SDRAMリフレッシュステータスレジスタ | DBRFSTS | R/W | H'FE80 004C | 32 | DDRck |
| DDRPAD周波数設定レジスタ | DBFREQ | R/W | H'FE80 0050 | 32 | DDRck |
| DDRPAD DIC, DIC, ODT, OCD設定レジスタ | DBDICODTOCD | R/W | H'FE80 0054 | 32 | DDRck |
| SDRAMモード設定レジスタ | DBMRCNT | W | H'FE80 0060 | 32 | DDRck |

レジスタの詳細については SH7785 ハードウェアマニュアル(RJJ09B0285)の「12 章 DDR2-SDRAM インタフェース(DBSC2)の 12.4 レジスタの説明」を参照下さい。

3.2 SDRAMに対するODT制御信号出力に関する注意点

DBSC2 から SDRAM に対して制御信号を出力する場合には、以下の注意点があります。

- SDRAM へ ODT 制御信号を出力する場合 CAS レイテンシーは DDR クロックで最低 4 サイクル必要です。
- ODT を DBDICODTOCD レジスタの ODT_EARLY ビットで ODT 制御信号を DDR クロックで 1 サイクル分手前から出力して延長する場合には、CAS レイテンシーは DDR クロックで最低 5 サイクル必要であり、かつ DBTR2 レジスタの RDWR ビットの設定値を SDRAM の仕様が要求する値に 1 加えたものとする必要があります。

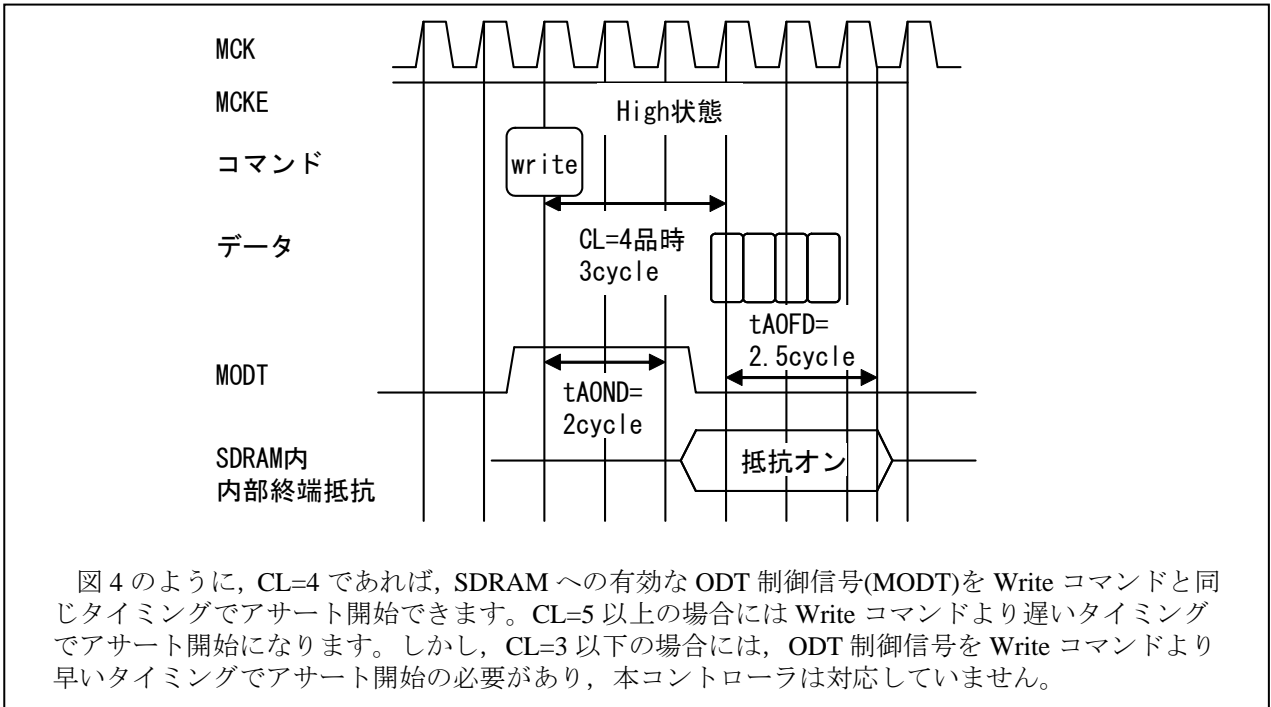


図 4 CL4 の時の ODT 制御信号

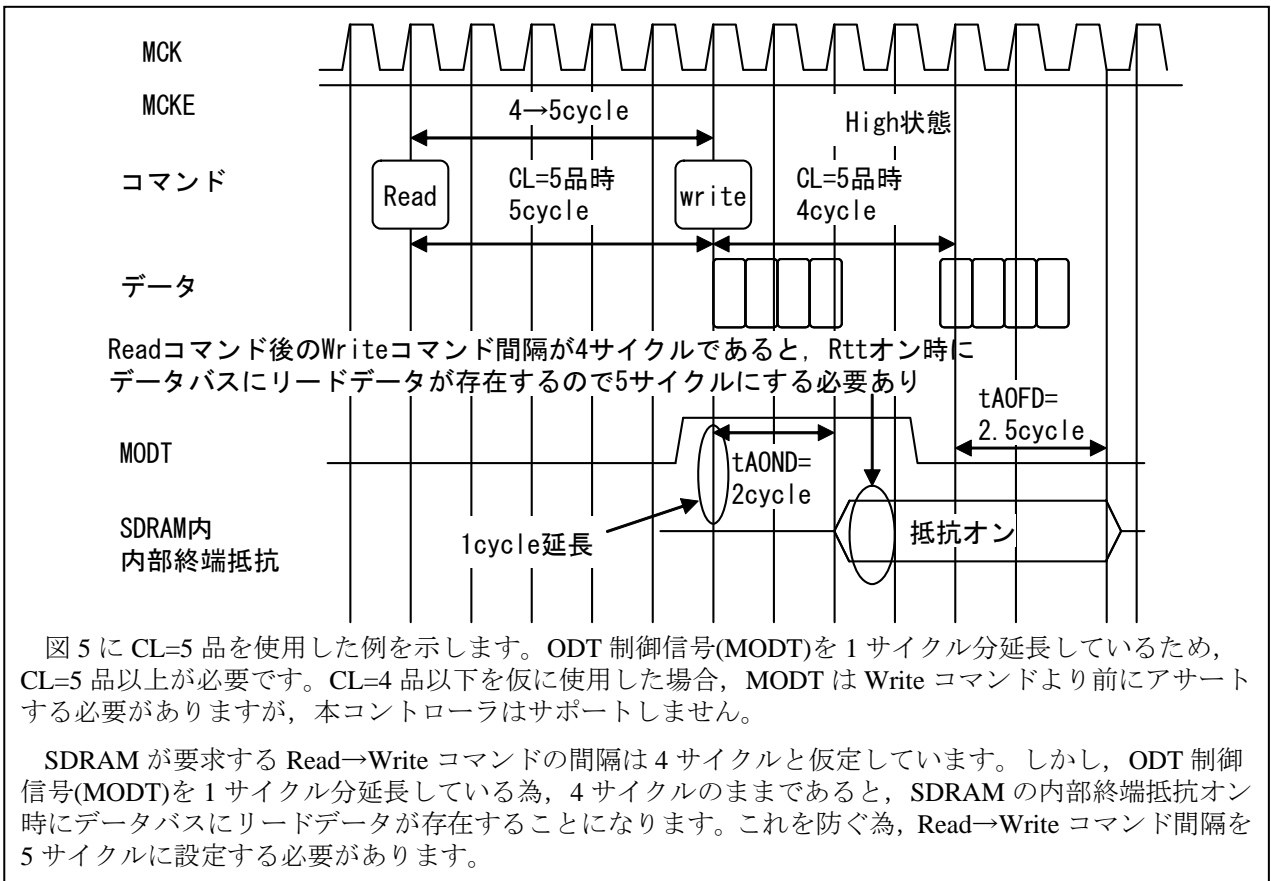


図 5 ODT 制御信号 1 サイクル延長時の注意点

4. 応用例の説明

4.1 R0P7785LC0011RLの概要

4.1.1 デバイス一覧

本アプリケーションノートの応用例で使用する R0P7785LC0011RL のデバイス一覧を表 7 に示します。

表 7 R0P7785LC0011RL 周辺デバイス

| 項目 | 内容 |
|-----------------|---|
| 制御 PLD | ALTERA 製 EPM240T100C5N ・ 8 ビットバスアクセス |
| USB2.0 コントローラ | ルネサスエレクトロニクス製 R8A66597FP#F0S ・ 16 ビットバスアクセス |
| SD コントローラ | D-broad 製 CG200-V2 ・ 32 ビットバスアクセス |
| 2D グラフィックコントローラ | Silicon Motion 製 SM107GX04LF0T-AA ・ 32 ビットバスアクセス |
| I2C コントローラ | NXP Semiconductors 製 PCA9564BS-T ・ 8 ビットバスアクセス |

4.1.2 メモリマップ

表 8 に R0P7785LC0011RL のメモリマップを示します。

表 8 R0P7785LC0011RL メモリマップ

| エリア | アドレス | 接続デバイス | バス幅 |
|-------------|-------------|-----------------------------|-------|
| 0 | H'0000_0000 | S29GL256P90TFIRI(64MB) | 32ビット |
| | H'03FF_FFFF | | |
| 1 | H'0400_0000 | EPM240T100C5N (16MB) | 8ビット |
| | H'05FF_FFFF | | |
| | H'0600_0000 | PCA9564BS-T (16MB) | |
| H'07FF_FFFF | | | |
| 2 | H'0800_0000 | EDE1108ACSE-6E-E (128MB) | 32ビット |
| | H'0BFF_FFFF | | |
| 3 | H'0C00_0000 | | |
| | H'0FFF_FFFF | | |
| 4 | H'1000_0000 | SM107GX04LF0T-AA (64MB) | 32ビット |
| | H'13FF_FFFF | | |
| 5 | H'1400_0000 | R8A66597FP#F0S (64MB) | 16ビット |
| | H'17FF_FFFF | | |
| 6 | H'1800_0000 | CG200-V2 (64MB) | 32ビット |
| | H'17FF_FFFF | | |

4.2 参考プログラムの説明

初期設定プログラムとして下記のソースプログラム内で設定を行っています。

例外/割り込みに使用するテーブルファイル等については必要に応じて設定の追加を行ってください。

- vhandler.src
- resetprg.c
- dbsct.c
- vecttbl.src
- vect.inc
- intprg.c
- lowlevelinit.src
- lowlevelinit.inc

(1) vhandler.src

例外 (リセット, 一般例外, 割り込み) が発生すると, 例外ハンドラ (vhandler.src) が実行されます。vhandler.src には各例外のハンドラ処理と LBSC, DBSC2 の初期化処理へジャンプする記述をしています。パワーオンリセットの場合は, リセットハンドラ (_Reset_handler) が起動されます。本アプリケーションノートで使用するリセットハンドラは, High-performance Embedded Workshop が自動生成するファイルとは異なり, LBSC, DBSC2 の初期化処理を追加しています。

(2) resetprg.c

resetprg.c は High-performance Embedded Workshop で自動生成される初期化関数ファイルを元に作成されており, vecttbl.src に登録する PowerON_Reset()関数を記述しています。

PowerON_Reset()関数はリセットハンドラより分岐する最初の関数であり, VBR (ベクタベースレジスタ) の設定, セクションコピーを行う _INITSCT()関数の呼び出し, キャッシュの有効化関数の呼び出しに続いてメイン関数を呼び出します。ステータスレジスタ (SR) は, 特権モード/ユーザモードの選択, 汎用レジスタバンク指定, 例外, 割り込みを管理します。システムの設計に基づいた設定をしてください。

(3) dbsct.c

dbsct.c には, プログラムを外部接続の SDRAM 上で実行させるために, ROM から RAM へのセクションのコピー情報を記載しています。_INITSCT()関数が本記載内容に従って, リセット直後にプログラムをコピーします。「SuperH RISC engine C/C++コンパイラパッケージ アプリケーションノート (RJJ05B0577)」を参考資料としてご参照ください。

(4) vecttbl.src

vecttbl.src は, 例外(リセット, 一般例外, 割り込み)が発生した時の例外ハンドラへの登録を記述しています。

(5) vect.inc

vect.inc は, vecttbl.src に登録する為の関数を記述しています。

(6) intprg.c

intprg.c は, 例外の割り込みが発生し, 例外ハンドラから登録した割り込み関数を記述しています。

(7)lowlevelinit.src

lowlevelinit.src は、LBSC、DBSC2 の初期設定を記述しています。lowlevelinit.src はリセットハンドラ (_Reset_handler)から呼び出されます。

(8)lowlevelinit.inc

lowlevelinit.inc は、lowlevelinit.src で使用する設定値を記述しています。

4.3 参考プログラムにおける設定内容

R0P7785LC0011RL に接続した、メモリ及び周辺デバイスにアクセスするために、LBSC,DBSC2 の設定を行います。

表 9 に参考プログラムでの設定を示します。

表 9 参考プログラムでの設定

| モジュール | 設定内容 |
|-------|--|
| LBSC | CS0 空間 : フラッシュメモリ(メモリタイプ SRAM) バス幅 32bit CS1 空間 : PLD(メモリタイプ SRAM) バス幅 8bit CS2,3 空間 : DDR2-SDRAM メモリ CS4 空間 : グラフィックコントローラ(メモリタイプ SRAM) バス幅 32bit CS5 空間 : USB コントローラ(メモリタイプ SRAM) バス幅 16bit CS6 空間 : SD コントローラ(メモリタイプ SRAM) バス幅 32bit |
| DBSC2 | 型名 : EDE1108ACSE-6E-E 容量(構成) : 1G ビット(128Mx8 ビット) 使用個数 : 4 個 CAS レイテンシ : 5 リフレッシュサイクル : 64ms, 平均リフレッシュ時間 : 7.8125 μ s バースト長 : 4/8(プログラマブル) ロウアドレス : A13~A0 カラムアドレス : A9~A0 プリチャージ : A10 で制御するオートプリジャージ/全バンクプリチャージ DriverStrength : nomal/weak(プログラマブル) ODT : 75 Ω |

4.4 参考プログラムフローチャート

図 6 にパワーオンリセットから main()関数にジャンプするまでの処理フローを示します。

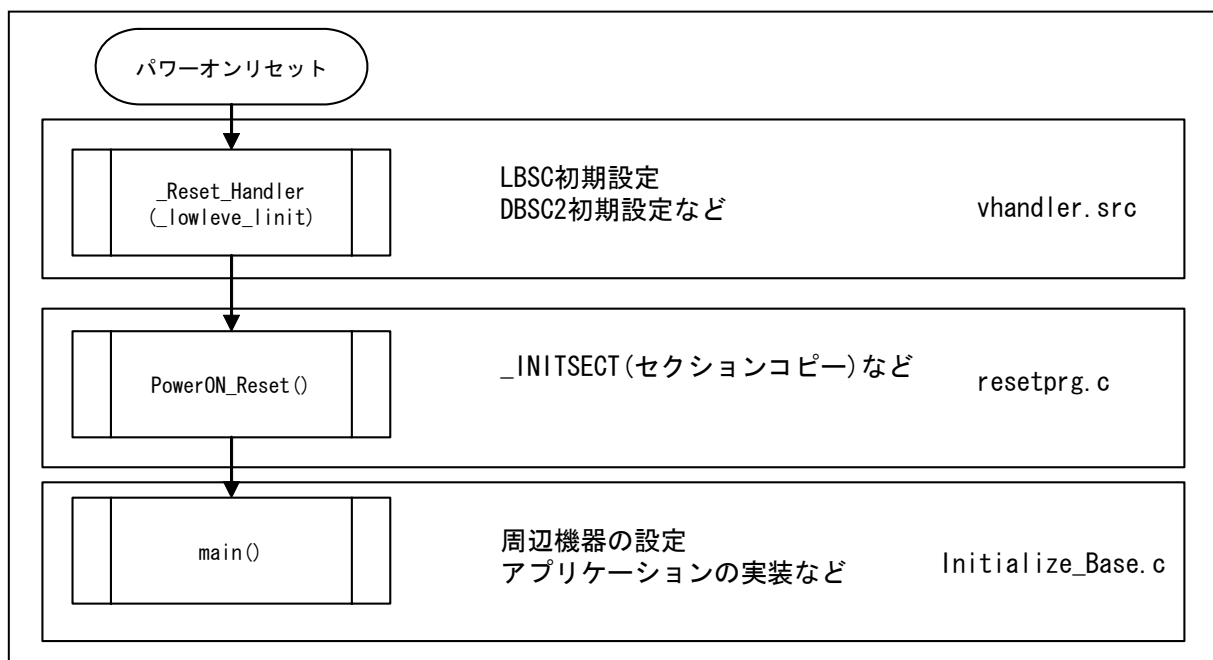


図 6 パワーオンリセットから main()関数までのフロー

図 7~11 に `_lowlevel_init()`関数のフローを示します。

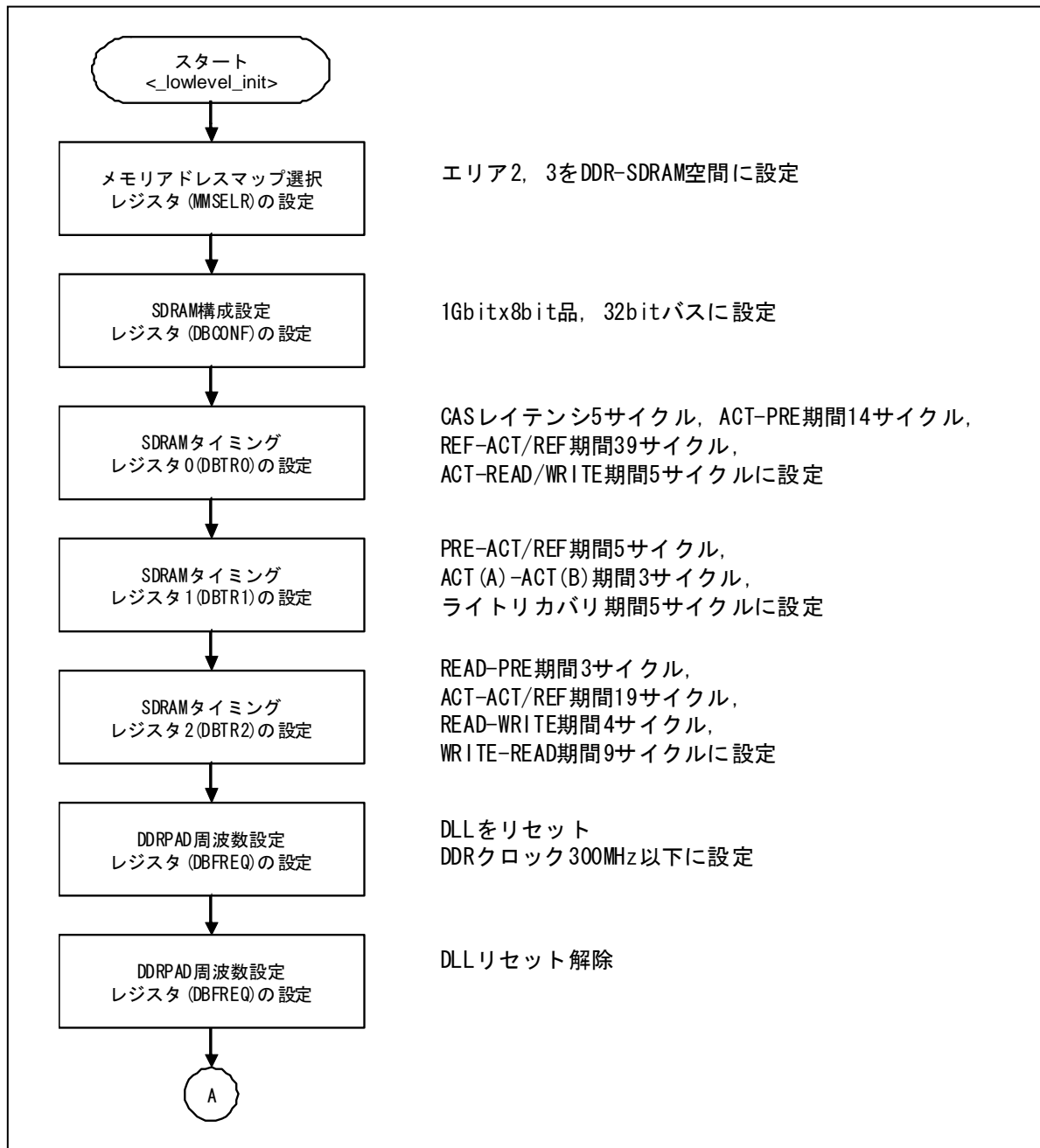


図 7 `_lowlevel_init()`関数フロー1

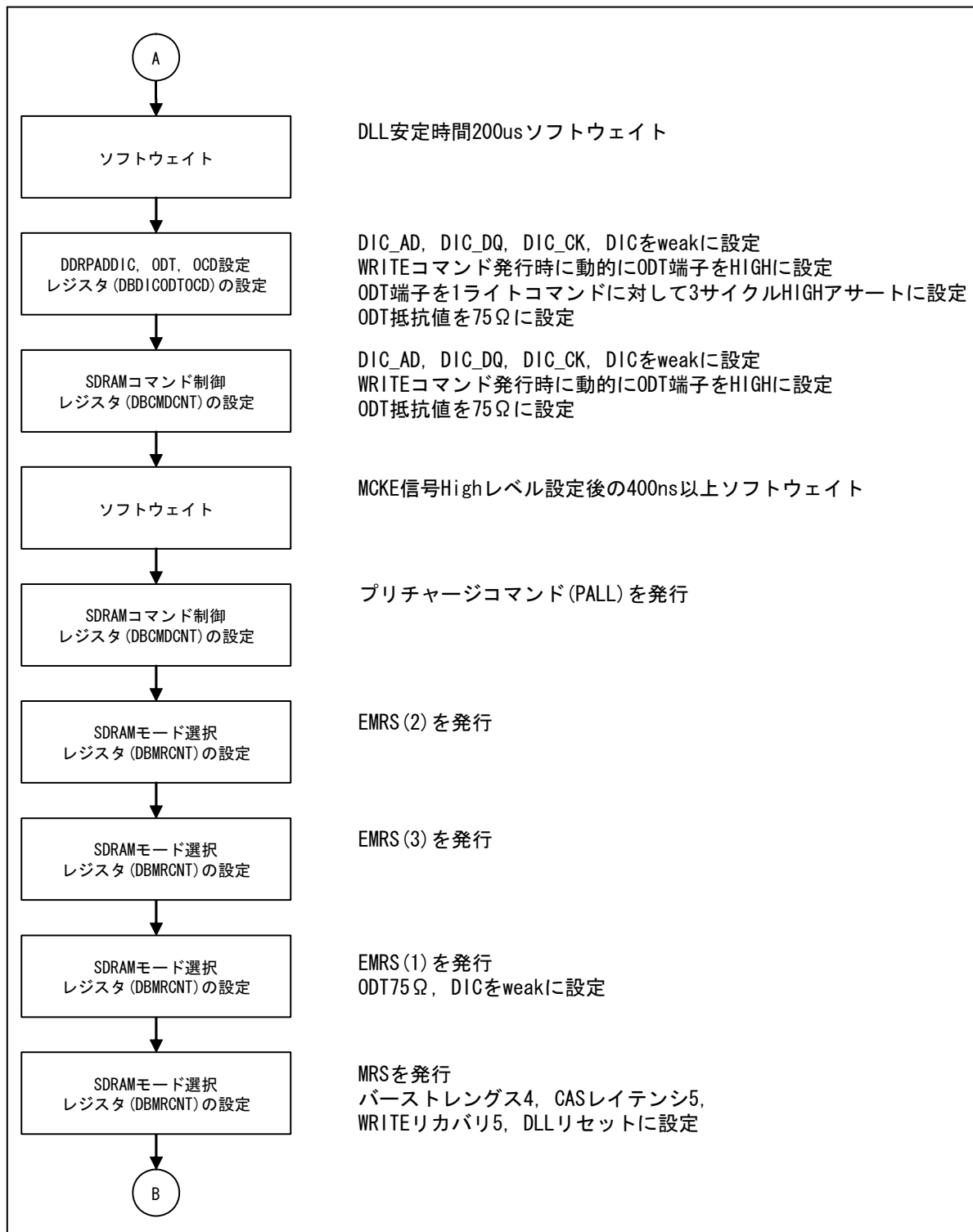


図 8 _lowlevel_init()関数フロー2

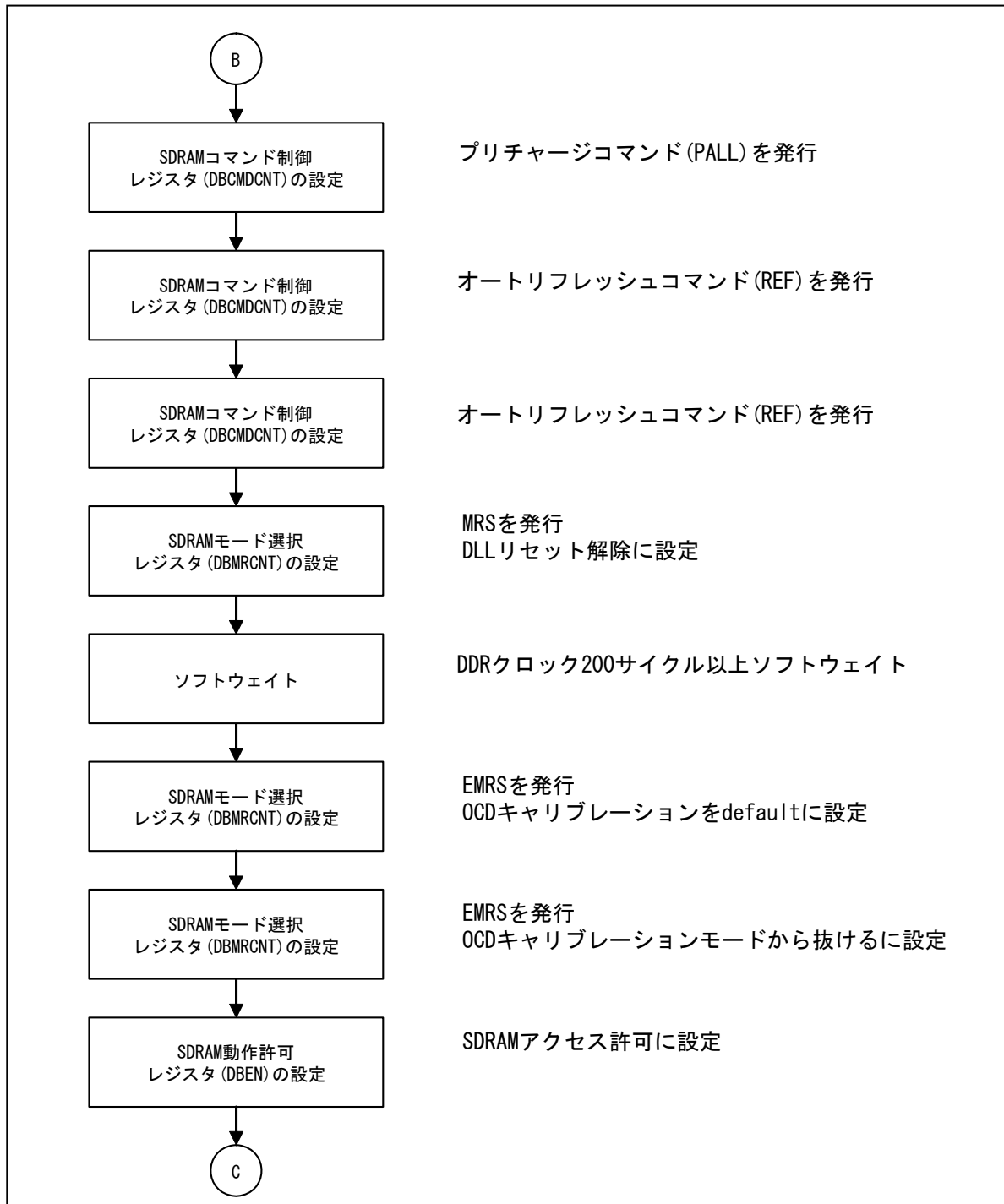


図 9 _lowlevel_init()関数フロー3

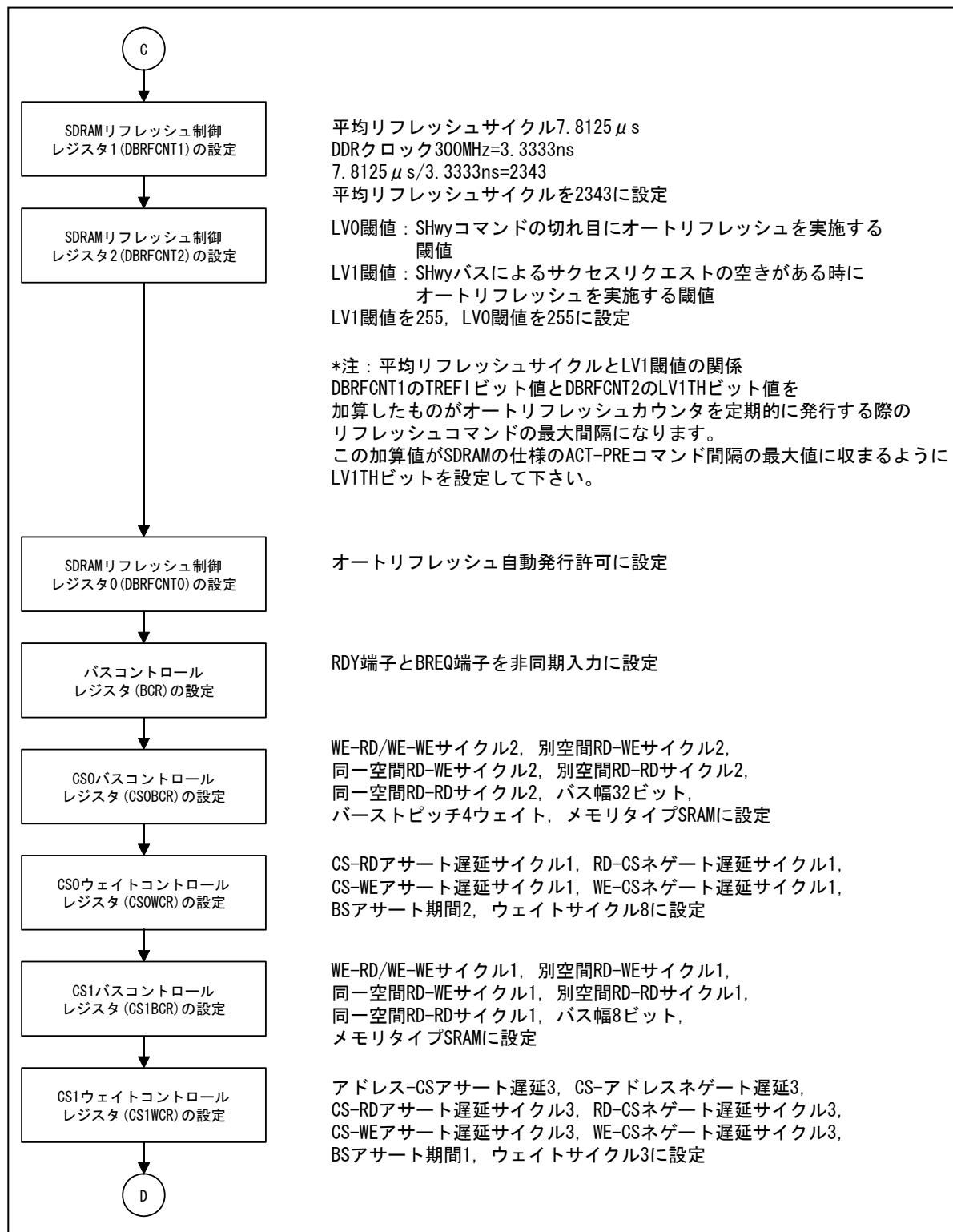


図 10 _lowlevel_init()関数フロー4

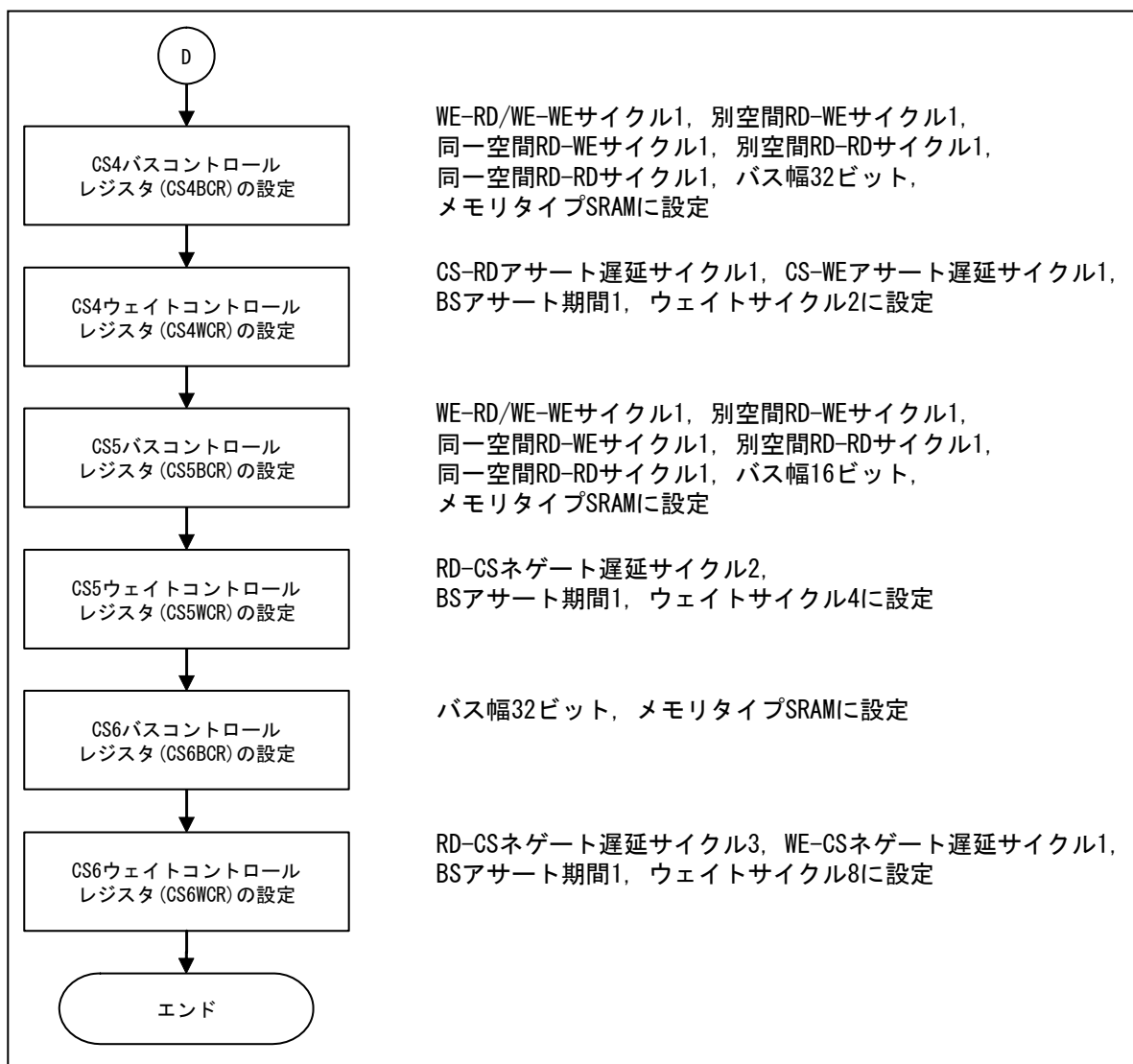


図 11 _lowlevel_init()関数フロー5

図 12 に PowerON_Reset()関数のフローを示します。

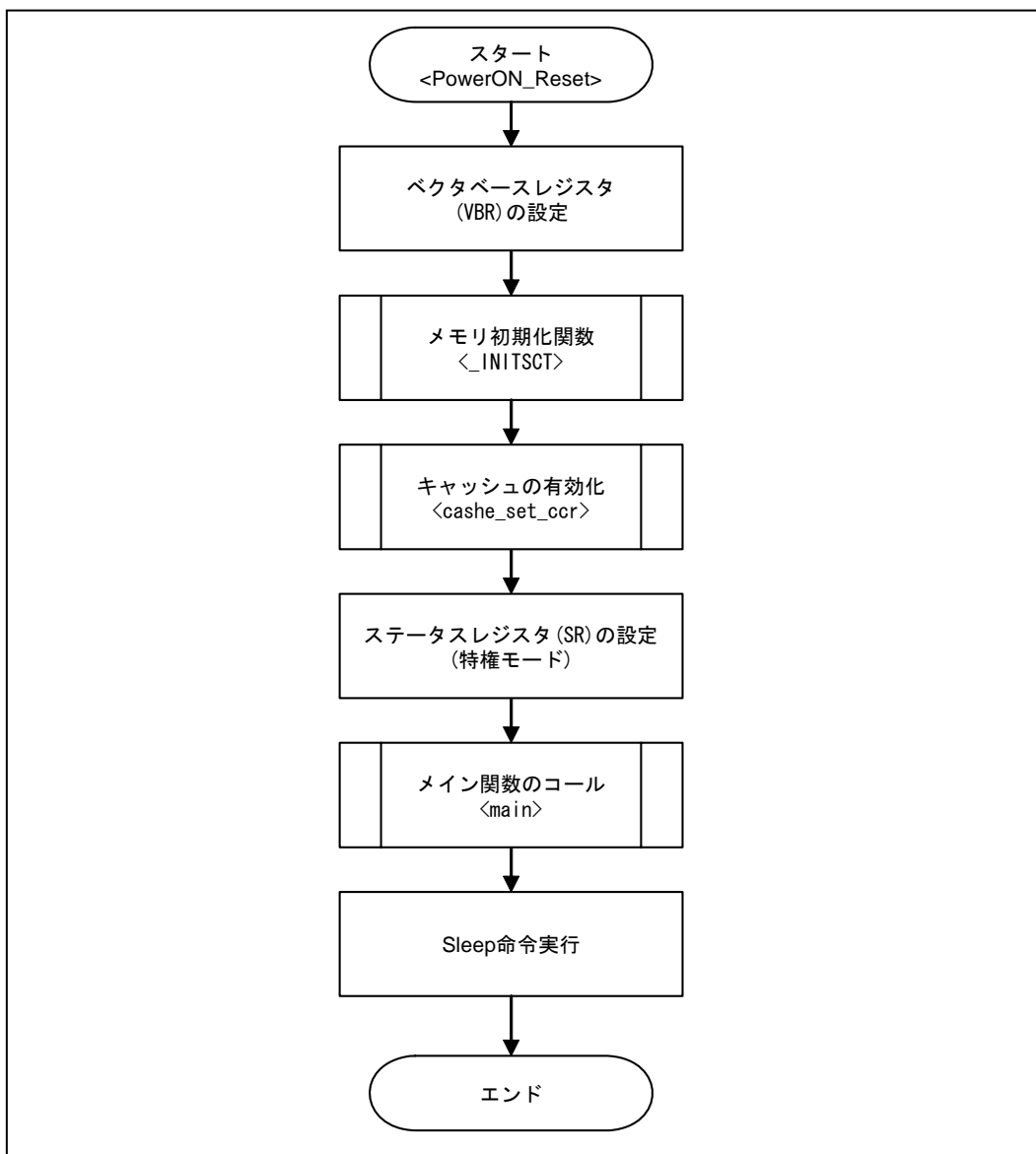


図 12 PowerON_Reset()関数フロー

5. 参考プログラム例

サンプルプログラムリスト”vhandler.src”

例外処理をしています。

```
001 ;/*****
002 ;* DISCLAIMER
003 ;
004 ;* This software is supplied by Renesas Electronics Corporation. and is only
005 ;* intended for use with Renesas products. No other uses are authorized.
006 ;
007 ;* This software is owned by Renesas Electronics Corporation. and is protected under
008 ;* all applicable laws, including copyright laws.
009 ;
010 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
014 ;* DISCLAIMED.
015 ;
016 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021 ;
022 ;* Renesas reserves the right, without notice, to make changes to this
023 ;* software and to discontinue the availability of this software.
024 ;* By using this software, you agree to the additional terms and
025 ;* conditions found by accessing the following link:
026 ;* http://www.renesas.com/disclaimer
027 ;/*****
028 ;/* Copyright (C) 2010. Renesas Electronics Corporation., All Rights Reserved.*/
029 ;/*"FILE COMMENT"***** Technical reference data *****
030 ;* System Name : SH7785 Sample Program
031 ;* File Name : vhandler.src
032 ;* Abstract : SH7785 初期設定例 Sample Program
033 ;* Version : Ver 1.00
034 ;* Device : SH7785
035 ;* Tool-Chain : High-performance Embedded Workshop (Version 4.07.00.007)
036 ;* : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 ;* OS : None
038 ;* H/W Platform : ルネサスエレクトロニクス製 SH-4A ボード 型番 ROP7785LC0011RL
039 ;* Description : SH7785 初期設定例のサンプルプログラムです。
040 ;* :
041 ;* :
042 ;* Operation :
043 ;* Limitation :
```

```

044 ;*
045 ;*****
046 ;* History      : 30.SEP.2010 Ver. 1.00 First Release
047 ;*"FILE COMMENT END"*****/
048 ;-----
049 ;
050 ; FILE          :vhandler.src
051 ; DATE          :Thu, May 13, 2010
052 ; DESCRIPTION   :Reset/Interrupt Handler
053 ; CPU TYPE      :Other
054 ;
055 ; This file is generated by Renesas Project Generator (Ver. 4.16).
056 ;
057 ;-----
058             .macro PUSH_EXP_BASE_REGA a, b
059     stc.l    ssr,@-r15           ; save ssr
060     stc.l    spc,@-r15           ; save spc
061     sts.l    pr,@-r15            ; save context registers
062     sts.l    fpscr,@-r15         ; save fpscr registers
063     stc.l    r7_bank,@-r15
064     stc.l    r6_bank,@-r15
065     stc.l    r5_bank,@-r15
066     stc.l    r4_bank,@-r15
067     stc.l    r3_bank,@-r15
068     stc.l    r2_bank,@-r15
069     stc.l    r1_bank,@-r15
070     stc.l    r0_bank,@-r15
071             .endm
072 ;
073
074
075     .include    "env.inc"
076     .include    "vect.inc"
077
078 ;
079 ILLEGALFPU_CODE:                .equ    H' 800
080 DUMMY_ILLEGALFPU_CODE:          .equ    H' 880
081 ILLEGALSLOTFPU_CODE:           .equ    H' 820
082 DUMMY_ILLEGALSLOTFPU_CODE:     .equ    H' 8A0
083 INT_NMI_CODE:                  .equ    H' 1C0
084 ;
085
086 IMASKclr: .equ    H' FFFFFFFF
087 RBBLclr:  .equ    H' CFFFFFFF
088 MDRBBLset: .equ    H' 70000000
089
090     .import    _RESET_Vectors

```

```

091     .import      _INT_Vectors
092     .import      _INT_MASK
093     .import      _lowlevel_init
094
095     .....
096 ;*          macro definition                      *;
097     .....
098         .macro PUSH_EXP_BASE_REG
099     stc.l  ssr, @-r15          ; save ssr
100     stc.l  spc, @-r15         ; save spc
101     sts.l  pr, @-r15          ; save context registers
102     sts.l  fpscr, @-r15       ; save fpscr registers
103     stc.l  r7_bank, @-r15
104     stc.l  r6_bank, @-r15
105     stc.l  r5_bank, @-r15
106     stc.l  r4_bank, @-r15
107     stc.l  r3_bank, @-r15
108     stc.l  r2_bank, @-r15
109     stc.l  r1_bank, @-r15
110     stc.l  r0_bank, @-r15
111         .endm
112 ;
113         .macro POP_EXP_BASE_REG
114     ldc.l  @r15+, r0_bank      ; recover registers
115     ldc.l  @r15+, r1_bank
116     ldc.l  @r15+, r2_bank
117     ldc.l  @r15+, r3_bank
118     ldc.l  @r15+, r4_bank
119     ldc.l  @r15+, r5_bank
120     ldc.l  @r15+, r6_bank
121     ldc.l  @r15+, r7_bank
122     lds.l  @r15+, fpscr
123     lds.l  @r15+, pr
124     ldc.l  @r15+, spc
125     ldc.l  @r15+, ssr
126         .endm
127 ;
128     .....
129 ;          reset
130     .....
131     .section      RSTHandler, code
132 _ResetHandler:
133
134
135         mov.l  #_lowlevel_init, r0
136         jsr          @r0

```



```

137          nop
138
139          mov.l  #EXPEVT, r0
140          mov.l  @r0, r0
141          shlr2  r0
142          shlr   r0
143          mov.l  #_RESET_Vectors, r1
144          add    r1, r0
145          mov.l  @r0, r0
146          jmp    @r0
147          nop
148 ;
149 ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
150 ;          exceptional interrupt                                     ;
151 ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
152          .section      INTHandler, code
153          .export       _INTHandlerPRG
154 _INTHandlerPRG:
155 _ExpHandler:
156          PUSH_EXP_BASE_REG
157 ;
158          mov.l  #EXPEVT, r0          ; set event address
159          mov.l  @r0, r1             ; set exception code
160
161 ;
162          mov.l  #ILLEGALFPU_CODE, r2          ; H' 800
163          cmp/eq r1, r2
164          bf     exp_01
165          mov.l  #DUMMY_ILLEGALFPU_CODE, r1    ; H' 800 -> H' 880
166          bra    exp_10
167          nop
168 exp_01:
169          mov.l  #ILLEGALSLOTFPU_CODE, r2      ; H' 820
170          cmp/eq r1, r2
171          bf     exp_10
172          mov.l  #DUMMY_ILLEGALSLOTFPU_CODE, r1 ; H' 820 -> H' 8A0
173 exp_10:
174
175 ;          mov.l  #_INT_Vectors, r0          ; set vector table address
176          add    #-(h' 40), r1              ; exception code - h' 40
177          shlr2  r1
178          shlr   r1
179          mov.l  @(r0, r1), r3              ; set interrupt function addr
180 ;
181          mov.l  #_INT_MASK, r0            ; interrupt mask table addr
182          shlr2  r1
183          mov.b  @(r0, r1), r1              ; interrupt mask

```

```

184          extu.b  r1,r1
185 ;
186          stc          sr,r0          ; save sr
187          mov.l  #(RBBLclr&IMASKclr),r2      ; RB,BL,mask clear data
188          and          r2,r0          ; clear mask data
189          or          r1,r0          ; set interrupt mask
190          ldc          r0,ssr          ; set current status
191 ;
192          ldc.l  r3,spc
193          mov.l  #__int_term,r0          ; set interrupt terminate
194          lds          r0,pr
195 ;
196          rte
197          nop
198 ;
199          .pool
200 ;
201 ::::::::::::::::::::::::::::::::::::::::::::::::::::
202 ;          Interrupt terminate
203 ;          ;
204 ::::::::::::::::::::::::::::::::::::::::::::::::::::
205          .align 4
206          __int_term:
207          mov.l  #MDRBBLset,r0      ; set MD,BL,RB
208          ldc.l  r0,sr          ;
209          POP_EXP_BASE_REG
210          rte          ; return
211 ;
212          .pool
213 ;
214 ::::::::::::::::::::::::::::::::::::::::::::::::::::
215 ;          TLB miss interrupt
216 ;          ;
217          .org  H'300
218          _TLBmissHandler:
219          PUSH_EXP_BASE_REG
220 ;
221          mov.l  #EXPEVT,r0          ; set event address
222          mov.l  @r0,r1          ; set exception code
223          mov.l  #_INT_Vectors,r0      ; set vector table address
224          add          #-(h'40),r1          ; exception code - h'40
225          shlr2  r1
226          shlr  r1
227          mov.l  @(r0,r1),r3          ; set interrupt function addr
228 ;
229          mov.l  #_INT_MASK,r0          ; interrupt mask table addr

```

```

230          shlr2   r1
231          mov.b   @(r0,r1),r1          ; interrupt mask
232          extu.b  r1,r1
233 ;
234          stc     sr,r0                ; save sr
235          mov.l   #(RBBLclr&IMASKclr),r2  ; RB,BL,mask clear data
236          and     r2,r0                ; clear mask data
237          or      r1,r0                ; set interrupt mask
238          ldc     r0,ssr               ; set current status
239 ;
240          ldc.l   r3,spc
241          mov.l   #_int_term,r0        ; set interrupt terminate
242          lds     r0,pr
243 ;
244          rte
245          nop
246 ;
247          .pool
248 ;
249 ::::::::::::::::::::::::::::::::::::::
250 ;      IRQ
251 ::::::::::::::::::::::::::::::::::::::
252          .org   H'500
253 _IRQHandler:
254          PUSH_EXP_BASE_REG
255 ;
256          mov.l   #INTEVT,r0           ; set event address
257          mov.l   @r0,r1               ; set exception code
258          mov.l   #_INT_Vectors,r0    ; set vector table address
259          add     #-(h'40),r1          ; exception code - h'40
260          shlr2  r1
261          shlr   r1
262          mov.l   @(r0,r1),r3          ; set interrupt function addr
263 ;
264          mov.l   #_INT_MASK,r0        ; interrupt mask table addr
265          shlr2  r1
266          mov.b   @(r0,r1),r1          ; interrupt mask
267          extu.b  r1,r1
268 ;
269          stc     sr,r0                ; save sr
270          mov.l   #(RBBLclr&IMASKclr),r2  ; RB,BL,mask clear data
271          and     r2,r0                ; clear mask data
272          or      r1,r0                ; set interrupt mask
273          ldc     r0,ssr               ; set current status
274 ;
275          ldc.l   r3,spc

```

```
276          mov.l  #_int_term, r0          ; set interrupt terminate
277          lds      r0, pr
278 ;
279          rte
280          nop
281 ;
282          .pool
283          .end
284
285
```

サンプルプログラムリスト”resetprg.src”

PowerON リセット関数です。

```
001 /*****
002 * DISCLAIMER
003
004 * This software is supplied by Renesas Electronics Corporation. and is only
005 * intended for use with Renesas products. No other uses are authorized.
006
007 * This software is owned by Renesas Electronics Corporation. and is protected under
008 * all applicable laws, including copyright laws.
009
010 * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
014 * DISCLAIMED.
015
016 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 * ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021
022 * Renesas reserves the right, without notice, to make changes to this
023 * software and to discontinue the availability of this software.
024 * By using this software, you agree to the additional terms and
025 * conditions found by accessing the following link:
026 * http://www.renesas.com/disclaimer
027 *****/
028 /* Copyright (C) 2010. Renesas Electronics Corporation., All Rights Reserved.*/
029 /*"FILE COMMENT"***** Technical reference data *****
030 * System Name : SH7785 Sample Program
031 * File Name : Initialize_Base.c
032 * Abstract : SH7785 初期設定例 Sample Program
033 * Version : Ver 1.00
034 * Device : SH7785
035 * Tool-Chain : High-performance Embedded Workshop (Version 4.07.00.007)
036 * : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 * OS : None
038 * H/W Platform : ルネサスエレクトロニクス製 SH-4A ボード 型番 R0P7785LC0011RL
039 * Description : SH7785 初期設定例のサンプルプログラムです。
040 * :
041 * :
042 * Operation :
043 * Limitation :
044 * :
045 *****/
```

```
046 * History      : 30.SEP.2010 Ver. 1.00 First Release
047 *"FILE COMMENT END"*****/
048 /*****/
049 /*                                                     */
050 /* FILE          :resetprg.c                            */
051 /* DATE          :Thu, May 13, 2010                     */
052 /* DESCRIPTION   :Reset Program                         */
053 /* CPU TYPE      :Other                                  */
054 /*                                                     */
055 /* This file is generated by Renesas Project Generator (Ver.4.16). */
056 /*                                                     */
057 /*****/
058
059
060
061 #include <machine.h>
062 #include <_h_c_lib.h>
063 //#include <stddef.h> // Remove the comment when you use
errno
064 //#include <stdlib.h> // Remove the comment when you use
rand()
065 #include "typedefine.h"
066 #include "stacksct.h"
067 #include "cache.h"
068
069
070 #define SR_Init 0x40000000
071 #define INT_OFFSET 0x100UL
072
073 #ifdef __cplusplus
074 extern "C" {
075 #endif
076 extern void INTHandlerPRG(void);
077 void PowerON_Reset(void);
078 void Manual_Reset(void);
079 void main(void);
080 #ifdef __cplusplus
081 }
082 #endif
083
084 #pragma section ResetPRG
085
086
087 #pragma entry PowerON_Reset
088
089 void PowerON_Reset(void)
090 {
091     set_vbr((void *)((_UINT) INTHandlerPRG - INT_OFFSET));
```

```
092
093
094  _INITSCT();
095
096 //      errno=0;                                // Remove the comment when you use
errno
097 //      srand((_UINT)1);                          // Remove the comment when you use
rand()
098 //      _s1ptr=NULL;                              // Remove the comment when you use strtok()
099  cache_set_ccr(D_CACHE_I_ON | D_CACHE_O_ON);
100
101  set_cr(SR_Init);
102
103  main();
104
105 //      _CLOSEALL();                               // Close I/O in the application(both SIM I/O
and hardware I/O)
106
107 //      _CALL_END();                               // Remove the comment when you use global
class object
108
109  sleep();
110 }
111
112 void Manual_Reset(void)
113 {
114 }
```

サンプルプログラムリスト” dbsect.c”

メモリの初期化を行っています。

```
01 /*****  
02 /*  
03 /* FILE      :dbsect.c      */  
04 /* DATE      :Thu, May 13, 2010 */  
05 /* DESCRIPTION :Setting of B,R Section */  
06 /* CPU TYPE   :Other        */  
07 /*  
08 /* This file is generated by Renesas Project Generator (Ver. 4.16). */  
09 /*  
10 /*****  
11  
12  
13  
14 #include "typedefine.h"  
15  
16  
17 #pragma section $DSEC  
18 static const struct {  
19     _UBYTE *rom_s;      /* 初期化データセクションのROM 上の先頭アドレス */  
20     _UBYTE *rom_e;      /* 初期化データセクションのROM 上の最終アドレス */  
21     _UBYTE *ram_s;      /* 初期化データセクションのRAM 上の先頭アドレス */  
22 } DTBL[] = {  
23     { __sectop("D"), __secend("D"), __sectop("R") }  
24 };  
25 #pragma section $BSEC  
26 static const struct {  
27     _UBYTE *b_s;        /* 未初期化データセクションの先頭アドレス */  
28     _UBYTE *b_e;        /* 未初期化データセクションの最終アドレス */  
29 } BTBL[] = {  
30     { __sectop("B"), __secend("B") }  
31 };
```


サンプルプログラムリスト” vecttbl.src”

割込み処理実行時の割込み優先度を設定しています。

```

001 ;/*****
002 ;* DISCLAIMER
003 ;
004 ;* This software is supplied by Renesas Electronics Corporation. and is only
005 ;* intended for use with Renesas products. No other uses are authorized.
006 ;
007 ;* This software is owned by Renesas Electronics Corporation. and is protected under
008 ;* all applicable laws, including copyright laws.
009 ;
010 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
014 ;* DISCLAIMED.
015 ;
016 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021 ;
022 ;* Renesas reserves the right, without notice, to make changes to this
023 ;* software and to discontinue the availability of this software.
024 ;* By using this software, you agree to the additional terms and
025 ;* conditions found by accessing the following link:
026 ;* http://www.renesas.com/disclaimer
027 ;*****/
028 ;/* Copyright (C) 2010. Renesas Electronics Corporation., All Rights Reserved.*/
029 ;/*"FILE COMMENT"***** Technical reference data *****
030 ;* System Name   : SH7785 Sample Program
031 ;* File Name    : vecttbl.src
032 ;* Abstract     : SH7785 初期設定例 Sample Program
033 ;* Version      : Ver 1.00
034 ;* Device       : SH7785
035 ;* Tool-Chain  : High-performance Embedded Workshop (Version 4.07.00.007)
036 ;*              : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 ;* OS           : None
038 ;* H/W Platform : ルネサスエレクトロニクス製 SH-4A ボード 型番 R0P7785LC0011RL
039 ;* Description  : SH7785 初期設定例のサンプルプログラムです。
040 ;*              :
041 ;*              :
042 ;* Operation    :
043 ;* Limitation   :
044 ;*              :
045 ;*****/

```

```

046 ;* History      : 30.SEP.2010 Ver. 1.00 First Release
047 ;*""FILE COMMENT END""*****/
048 ;-----
049 ;
050 ; FILE          :vecttbl.src
051 ; DATE          :Thu, May 13, 2010
052 ; DESCRIPTION  :Initialize of Vector Table
053 ; CPU TYPE     :Other
054 ;
055 ; This file is generated by Renesas Project Generator (Ver.4.16).
056 ;
057 ;-----
058 ;/*****/
059 ;*
060 ;* Device       : SH-4A/SH7785
061 ;*
062 ;* File Name    : vecttbl.src
063 ;*
064 ;* Abstract     : Initialize of Vector Table.
065 ;*
066 ;* History      : 1.00 (2010-09-30) [Hardware Manual Revision : 1.00]
067 ;*
068 ;* Copyright(c) 2010 Renesas Electronics Corporation.
069 ;*              And Renesas Solutions Corp.,All Rights Reserved.
070 ;*
071 ;*****/
072
073 ; .include      "config.h"
074
075 ; .include      "vect.inc"
076
077 ; .section      VECTTBL,data
078 ; .export      _RESET_Vectors
079
080 _RESET_Vectors:
081 ;<<VECTOR DATA START (POWER ON RESET)>>
082 ;H'000 Power On Reset (Hitachi-UDI RESET)
083 ; .data.l      _PowerON_Reset
084 ;<<VECTOR DATA END (POWER ON RESET)>>
085 ;<<VECTOR DATA START (MANUAL RESET)>>
086 ;H'020 Manual Reset
087 ; .data.l      _Manual_Reset
088 ;<<VECTOR DATA END (MANUAL RESET)>>
089 ; Reserved
090 ; .datab.l      8,H'00000000
091 ;<<VECTOR DATA START (TLB RESET)>>
092 ;H'140 TLB Reset (DATA TLB Reset)

```

```

093     .data |      _TLB_Reset
094 :<<VECTOR DATA END (TLB RESET)>>
095
096     .section  INTTBL,data
097     .export   _INT_Vectors
098 _INT_Vectors:
099         ; H' 040      Data TLB miss exception(read)
100         .data |      _INT_TLB_MISS_READ_EXP
101         ; H' 060      Data TLB miss exception(write)
102         .data |      _INT_TLB_MISS_WRITE_EXP
103         ; H' 080      Initial page write exception
104         .data |      _INT_TLB_INIT_PAGE_EXP
105         ; H' 0A0      Data TLB protection violation exception (read)
106         .data |      _INT_TLB_PROTECT_READ_EXP
107         ; H' 0C0      Data TLB protection violation exception (write)
108         .data |      _INT_TLB_PROTECT_WRITE_EXP
109         ; H' 0E0      Data address error(read)
110         .data |      _INT_ADR_ERROR_READ
111         ; H' 100      Data address error(write)
112         .data |      _INT_ADR_ERROR_WRITE
113         ; H' 120      FPU exception
114         .data |      _INT_FPU_EXP
115         ; H' 140      Instruction TLB multiple-hit exception
116         .data |      _TLB_Reset
117         ; H' 160      Unconditional trap(TRAPA)
118         .data |      _INT_TRAP
119         ; H' 180      General illegal instruction exception
120         .data |      _INT_ILLEGAL_INST_EXP
121         ; H' 1A0      Slot illegal instruction exception
122         .data |      _INT_ILLEGAL_SLOT_EXP
123 :EXTERNAL INTERRUPT
124         ; H' 1C0      NMI
125         .data |      _INT_NMI
126         ; H' 1E0      USER_BREAK
127         .data |      _INT_USER_BREAK
128
129         ; H' 200      IRL30_LEVEL15_IRQ7
130         .data |      _INT_IRL30_LEVEL15_IRQ7
131         ; H' 220      IRL30_LEVEL14
132         .data |      _INT_IRL30_LEVEL14
133         ; H' 240      IRL30_LEVEL13_IRQ0
134         .data |      _INT_IRL30_LEVEL13_IRQ0
135         ; H' 260      IRL30_LEVEL12
136         .data |      _INT_IRL30_LEVEL12
137         ; H' 280      IRL30_LEVEL11_IRQ1
138         .data |      _INT_IRL30_LEVEL11_IRQ1
139         ; H' 2A0      IRL30_LEVEL10

```

| | | |
|--------------|-------------|------------------------|
| 140 | . data. I | _INT_IRL30_LEVEL10 |
| 141 | ; H' 2C0 | IRL30_LEVEL9_IRQ2 |
| 142 | . data. I | _INT_IRL30_LEVEL9_IRQ2 |
| 143 | ; H' 2E0 | IRL30_LEVEL8 |
| 144 | . data. I | _INT_IRL30_LEVEL8 |
| 145 | ; H' 300 | IRL30_LEVEL7_IRQ3 |
| 146 | . data. I | _INT_IRL30_LEVEL7_IRQ3 |
| 147 | ; H' 320 | IRL30_LEVEL6 |
| 148 | . data. I | _INT_IRL30_LEVEL6 |
| 149 | ; H' 340 | IRL30_LEVEL5_IRQ4 |
| 150 | . data. I | _INT_IRL30_LEVEL5_IRQ4 |
| 151 | ; H' 360 | IRL30_LEVEL4 |
| 152 | . data. I | _INT_IRL30_LEVEL4 |
| 153 | ; H' 380 | IRL30_LEVEL3_IRQ5 |
| 154 | . data. I | _INT_IRL30_LEVEL3_IRQ5 |
| 155 | ; H' 3A0 | IRL30_LEVEL2 |
| 156 | . data. I | _INT_IRL30_LEVEL2 |
| 157 | ; H' 3C0 | IRL30_LEVEL1_IRQ6 |
| 158 | . data. I | _INT_IRL30_LEVEL1_IRQ6 |
| 159 | ;H' 3E0-540 | Reserved |
| 160 | . datab. I | 12, H' 00000000 |
| 161 ;WDT | | |
| 162 | ;H' 560 | WDT_ITI |
| 163 | . data. I | _INT_WDT_ITI |
| 164 ;TMU-ch0 | | |
| 165 | ;H' 580 | TMU_TUNI0 |
| 166 | . data. I | _INT_TMU_TUNI0 |
| 167 ;TMU-ch1 | | |
| 168 | ;H' 5A0 | TMU_TUNI1 |
| 169 | . data. I | _INT_TMU_TUNI1 |
| 170 ;TMU-ch2 | | |
| 171 | ;H' 5C0 | TMU_TUNI2 |
| 172 | . data. I | _INT_TMU_TUNI2 |
| 173 | ;H' 5E0 | TMU_TICPI2 |
| 174 | . data. I | _INT_TMU_TICPI2 |
| 175 ;H-UDI | | |
| 176 | ;H' 600 | H-UDI I |
| 177 | . data. I | _INT_H_UDI I |
| 178 ;DMAC(0) | | |
| 179 | ;H' 620 | DMINT0 |
| 180 | . data. I | _INT_DMAC_DMINT0 |
| 181 | ;H' 640 | DMINT1 |
| 182 | . data. I | _INT_DMAC_DMINT1 |
| 183 | ;H' 660 | DMINT2 |
| 184 | . data. I | _INT_DMAC_DMINT2 |
| 185 | ;H' 680 | DMINT3 |
| 186 | . data. I | _INT_DMAC_DMINT3 |

| | | |
|-----|-------------|-------------------|
| 187 | :H' 6A0 | DMINT4 |
| 188 | .data. l | _INT_DMAC_DMINT4 |
| 189 | :H' 6C0 | DMINT5 |
| 190 | .data. l | _INT_DMAC_DMINT5 |
| 191 | :H' 6E0 | DMAE0 (ch0-5) |
| 192 | .data. l | _INT_DMAC_DMAE0 |
| 193 | :SCIF (0) | |
| 194 | :H' 700 | SCIF_ERI0 |
| 195 | .data. l | _INT_SCIF_ERI0 |
| 196 | :H' 720 | SCIF_RXI0 |
| 197 | .data. l | _INT_SCIF_RXI0 |
| 198 | :H' 740 | SCIF_BRI0 |
| 199 | .data. l | _INT_SCIF_BRI0 |
| 200 | :H' 760 | SCIF_TXI0 |
| 201 | .data. l | _INT_SCIF_TXI0 |
| 202 | :SCIF (1) | |
| 203 | :H' 780 | SCIF_ERI1 |
| 204 | .data. l | _INT_SCIF_ERI1 |
| 205 | :H' 7A0 | SCIF_RXI1 |
| 206 | .data. l | _INT_SCIF_RXI1 |
| 207 | :H' 7C0 | SCIF_BRI1 |
| 208 | .data. l | _INT_SCIF_BRI1 |
| 209 | :H' 7E0 | SCIF_TXI1 |
| 210 | .data. l | _INT_SCIF_TXI1 |
| 211 | :H' 800-860 | Reserved |
| 212 | .datab. l | 4, H' 00000000 |
| 213 | :DMAC (1) | |
| 214 | :H' 880 | DMINT6 |
| 215 | .data. l | _INT_DMAC_DMINT6 |
| 216 | :H' 8A0 | DMINT7 |
| 217 | .data. l | _INT_DMAC_DMINT7 |
| 218 | :H' 8C0 | DMINT8 |
| 219 | .data. l | _INT_DMAC_DMINT8 |
| 220 | :H' 8E0 | DMINT9 |
| 221 | .data. l | _INT_DMAC_DMINT9 |
| 222 | :H' 900 | DMINT10 |
| 223 | .data. l | _INT_DMAC_DMINT10 |
| 224 | :H' 920 | DMINT11 |
| 225 | .data. l | _INT_DMAC_DMINT11 |
| 226 | :H' 940 | DMAE1 (ch6-11) |
| 227 | .data. l | _INT_DMAC_DMAE0 |
| 228 | :HSP1 | |
| 229 | :H' 960 | HSP1_SPI1 |
| 230 | .data. l | _INT_HSP1_SPI1 |
| 231 | :SCIF (2) | |
| 232 | :H' 980 | SCIF_SCIF12 |
| 233 | .data. l | _INT_SCIF_SCIF12 |

| | | | |
|-----|-----------|----------|--------------------|
| 234 | :SCIF(3) | | |
| 235 | | :H' 9A0 | SCIF_SCIF13 |
| 236 | | .data.l | _INT_SCIF_SCIF13 |
| 237 | :SCIF(4) | | |
| 238 | | :H' 9C0 | SCIF_SCIF14 |
| 239 | | .data.l | _INT_SCIF_SCIF14 |
| 240 | :SCIF(5) | | |
| 241 | | :H' 9E0 | SCIF_SCIF15 |
| 242 | | .data.l | _INT_SCIF_SCIF15 |
| 243 | :PGIC(0) | | |
| 244 | | :H' A00 | PCI_PCISERR |
| 245 | | .data.l | _INT_PCI_PCISERR |
| 246 | :PGIC(1) | | |
| 247 | | :H' A20 | PCI_PCIINTA |
| 248 | | .data.l | _INT_PCI_PCIINTA |
| 249 | :PGIC(2) | | |
| 250 | | :H' A40 | PCI_PCIINTB |
| 251 | | .data.l | _INT_PCI_PCIINTB |
| 252 | :PGIC(3) | | |
| 253 | | :H' A60 | PCI_PCIINTC |
| 254 | | .data.l | _INT_PCI_PCIINTC |
| 255 | :PGIC(4) | | |
| 256 | | :H' A80 | PCI_PCIINTD |
| 257 | | .data.l | _INT_PCI_PCIINTD |
| 258 | :PGIC(5) | | |
| 259 | | :H' AA0 | PCI_PCERR |
| 260 | | .data.l | _INT_PCI_PCERR |
| 261 | | :H' AC0 | PCI_PCIPWD3_1 |
| 262 | | .data.l | _INT_PCI_PCIPWD3_1 |
| 263 | | :H' AE0 | PCI_PCIPWD0 |
| 264 | | .data.l | _INT_PCI_PCIPWD0 |
| 265 | :IRL(7-4) | | |
| 266 | | : H' B00 | IRL74_LEVEL15 |
| 267 | | .data.l | _INT_IRL74_LEVEL15 |
| 268 | | : H' B20 | IRL74_LEVEL14 |
| 269 | | .data.l | _INT_IRL74_LEVEL14 |
| 270 | | : H' B40 | IRL74_LEVEL13 |
| 271 | | .data.l | _INT_IRL74_LEVEL13 |
| 272 | | : H' B60 | IRL74_LEVEL12 |
| 273 | | .data.l | _INT_IRL74_LEVEL12 |
| 274 | | : H' B80 | IRL74_LEVEL11 |
| 275 | | .data.l | _INT_IRL74_LEVEL11 |
| 276 | | : H' BA0 | IRL74_LEVEL10 |
| 277 | | .data.l | _INT_IRL74_LEVEL10 |
| 278 | | : H' BC0 | IRL74_LEVEL9 |
| 279 | | .data.l | _INT_IRL74_LEVEL9 |
| 280 | | : H' BE0 | IRL74_LEVEL8 |

| | | |
|-----|-----------|-------------------|
| 281 | . data. I | _INT_IRL74_LEVEL8 |
| 282 | ; H' C00 | IRL74_LEVEL7 |
| 283 | . data. I | _INT_IRL74_LEVEL7 |
| 284 | ; H' C20 | IRL74_LEVEL6 |
| 285 | . data. I | _INT_IRL74_LEVEL6 |
| 286 | ; H' C40 | IRL74_LEVEL5 |
| 287 | . data. I | _INT_IRL74_LEVEL5 |
| 288 | ; H' C60 | IRL74_LEVEL4 |
| 289 | . data. I | _INT_IRL74_LEVEL4 |
| 290 | ; H' C80 | IRL74_LEVEL3 |
| 291 | . data. I | _INT_IRL74_LEVEL3 |
| 292 | ; H' CA0 | IRL74_LEVEL2 |
| 293 | . data. I | _INT_IRL74_LEVEL2 |
| 294 | ; H' CC0 | IRL74_LEVEL1 |
| 295 | . data. I | _INT_IRL74_LEVEL1 |
| 296 | :SIOF | |
| 297 | ;H' CE0 | SIOF_SIOF I |
| 298 | . data. I | _INT_SIOF_SIOF I |
| 299 | :MMCIF | |
| 300 | ;H' D00 | MMCIF_FSTAT |
| 301 | . data. I | _INT_MMCIF_FSTAT |
| 302 | ;H' D20 | MMCIF_TRAN |
| 303 | . data. I | _INT_MMCIF_TRAN |
| 304 | ;H' D40 | MMCIF_ERR |
| 305 | . data. I | _INT_MMCIF_ERR |
| 306 | ;H' D60 | MMCIF_FRDY |
| 307 | . data. I | _INT_MMCIF_FRDY |
| 308 | :DU | |
| 309 | ;H' D80 | DU_DU I |
| 310 | . data. I | _INT_DU_DU I |
| 311 | :GDTA | |
| 312 | ;H' DA0 | GDTA_GACL I |
| 313 | . data. I | _INT_GDTA_GACL I |
| 314 | ;H' DC0 | GDTA_GAMC I |
| 315 | . data. I | _INT_GDTA_GAMC I |
| 316 | ;H' DE0 | GDTA_GAER I |
| 317 | . data. I | _INT_GDTA_GAER I |
| 318 | :TMU-ch3 | |
| 319 | ;H' E00 | TMU_TUNI3 |
| 320 | . data. I | _INT_TMU_TUNI3 |
| 321 | :TMU-ch4 | |
| 322 | ;H' E20 | TMU_TUNI4 |
| 323 | . data. I | _INT_TMU_TUNI4 |
| 324 | :TMU-ch5 | |
| 325 | ;H' E40 | TMU_TUNI5 |
| 326 | . data. I | _INT_TMU_TUNI5 |
| 327 | ;H' E60 | Reserved |

```

328          .data. l      H' 00000000
329 ;SSI-ch0
330          ;H' E80      SSI_SSI10
331          .data. l      _INT_SSI_SSI10
332 ;SSI-ch1
333          ;H' EA0      SSI_SSI11
334          .data. l      _INT_SSI_SSI11
335 ;HAC-ch0
336          ;H' EC0      HAC_HAC10
337          .data. l      _INT_HAC_HAC10
338 ;HAC-ch1
339          ;H' EE0      HAC_HAC11
340          .data. l      _INT_HAC_HAC11
341 ;FLCTL
342          ;H' F00      FLCTL_FLSTE
343          .data. l      _INT_FLCTL_FLSTE
344          ;H' F20      FLCTL_FLTEND
345          .data. l      _INT_FLCTL_FLTEND
346          ;H' F40      FLCTL_FLTRQ0
347          .data. l      _INT_FLCTL_FLTRQ0
348          ;H' F60      FLCTL_FLTRQ1
349          .data. l      _INT_FLCTL_FLTRQ1
350 ;GPIO
351          ;H' F80      GPIO_GPIO10
352          .data. l      _INT_GPIO_GPIO10
353          ;H' FA0      GPIO_GPIO11
354          .data. l      _INT_GPIO_GPIO11
355          ;H' FC0      GPIO_GPIO12
356          .data. l      _INT_GPIO_GPIO12
357          ;H' FE0      GPIO_GPIO13
358          .data. l      _INT_GPIO_GPIO13
359
360          .export      _INT_MASK
361 _INT_MASK:
362          ; interrupt priority mask level (31~0)
363
364          ;H' 040      Data TLB miss exception(read)
365          .data. b      H' F0
366          ;H' 060      Data TLB miss exception(write)
367          .data. b      H' F0
368          ;H' 080      Initial page write exception
369          .data. b      H' F0
370          ;H' 0A0      Data TLB protection violation exception (read)
371          .data. b      H' F0
372          ;H' 0C0      Data TLB protection violation exception (write)
373          .data. b      H' F0
374          ;H' 0E0      Data address error (read)

```


| | | |
|-----|---------------------|--|
| 375 | . data. b | H' F0 |
| 376 | :H' 100 | Data address error (write) |
| 377 | . data. b | H' F0 |
| 378 | :H' 120 | FPU exception |
| 379 | . data. b | H' F0 |
| 380 | :H' 140 | Instruction TLB multiple-hit exception |
| 381 | . data. b | H' F0 |
| 382 | :H' 160 | TRAPA |
| 383 | . data. b | H' F0 |
| 384 | :H' 180 | ILLEGAL_INST |
| 385 | . data. b | H' F0 |
| 386 | :H' 1A0 | ILLEGAL_SLOT |
| 387 | . data. b | H' F0 |
| 388 | ;EXTERNAL INTERRUPT | |
| 389 | :H' 1c0 | NMI |
| 390 | . data. b | H' F0 |
| 391 | :H' 1E0 | USER_BREAK |
| 392 | . data. b | H' F0 |
| 393 | | |
| 394 | : H' 200-3C0 | IRL_IRQ7-0 |
| 395 | . datab. b | 15, H' F0 |
| 396 | | |
| 397 | :H' 3E0-540 | Reserved |
| 398 | . datab. b | 12, H' F0 |
| 399 | ;WDT | |
| 400 | :H' 560 | WDT_ITI |
| 401 | . data. b | H' F0 |
| 402 | ;TMU-ch0 | |
| 403 | :H' 580 | TMU_TUNI0 |
| 404 | . data. b | H' F0 |
| 405 | ;TMU-ch1 | |
| 406 | :H' 5A0 | TMU_TUNI1 |
| 407 | . data. b | H' F0 |
| 408 | ;TMU-ch2 | |
| 409 | :H' 5C0 | TMU_TUNI2 |
| 410 | . data. b | H' F0 |
| 411 | :H' 5E0 | TMU_TICPI2 |
| 412 | . data. b | H' F0 |
| 413 | ;H-UDI | |
| 414 | :H' 600 | H-UDI1 |
| 415 | . data. b | H' F0 |
| 416 | ;DMAC(0) | |
| 417 | :H' 620 | DMINT0 |
| 418 | . data. b | H' F0 |
| 419 | :H' 640 | DMINT1 |
| 420 | . data. b | H' F0 |
| 421 | :H' 660 | DMINT2 |

| | | |
|--------------|-------------|----------------|
| 422 | . data. b | H' F0 |
| 423 | ;H' 680 | DMINT3 |
| 424 | . data. b | H' F0 |
| 425 | ;H' 6A0 | DMINT4 |
| 426 | . data. b | H' F0 |
| 427 | ;H' 6C0 | DMINT5 |
| 428 | . data. b | H' F0 |
| 429 | ;H' 6E0 | DMAE0 (ch0-5) |
| 430 | . data. b | H' F0 |
| 431 ;SCIF(0) | | |
| 432 | ;H' 700 | SCIF_ERI0 |
| 433 | . data. b | H' F0 |
| 434 | ;H' 720 | SCIF_RXI0 |
| 435 | . data. b | H' F0 |
| 436 | ;H' 740 | SCIF_BRI0 |
| 437 | . data. b | H' F0 |
| 438 | ;H' 760 | SCIF_TXI0 |
| 439 | . data. b | H' F0 |
| 440 ;SCIF(1) | | |
| 441 | ;H' 780 | SCIF_ERI1 |
| 442 | . data. b | H' F0 |
| 443 | ;H' 7A0 | SCIF_RXI1 |
| 444 | . data. b | H' F0 |
| 445 | ;H' 7C0 | SCIF_BRI1 |
| 446 | . data. b | H' F0 |
| 447 | ;H' 7E0 | SCIF_TXI1 |
| 448 | . data. b | H' F0 |
| 449 | ;H' 800-860 | Reserved |
| 450 | . datab. b | 4, H' F0 |
| 451 ; | . data. b | H' F0 |
| 452 ;DMAC(1) | | |
| 453 | ;H' 880 | DMINT6 |
| 454 | . data. b | H' F0 |
| 455 | ;H' 8A0 | DMINT7 |
| 456 | . data. b | H' F0 |
| 457 | ;H' 8C0 | DMINT8 |
| 458 | . data. b | H' F0 |
| 459 | ;H' 8E0 | DMINT9 |
| 460 | . data. b | H' F0 |
| 461 | ;H' 900 | DMINT10 |
| 462 | . data. b | H' F0 |
| 463 | ;H' 920 | DMINT11 |
| 464 | . data. b | H' F0 |
| 465 | ;H' 940 | DMAE1 (ch6-11) |
| 466 | . data. b | H' F0 |
| 467 ;HSPI | | |
| 468 | ;H' 960 | HSPI_SPI1 |

| | | |
|---------------|--------------|---------------|
| 469 | . data. b | H' F0 |
| 470 ;SCIF(2) | | |
| 471 | :H' 980 | SCIF_SCIF12 |
| 472 | . data. b | H' F0 |
| 473 ;SCIF(3) | | |
| 474 | :H' 9A0 | SCIF_SCIF13 |
| 475 | . data. b | H' F0 |
| 476 ;SCIF(4) | | |
| 477 | :H' 9C0 | SCIF_SCIF14 |
| 478 | . data. b | H' F0 |
| 479 ;SCIF(5) | | |
| 480 | :H' 9E0 | SCIF_SCIF15 |
| 481 | . data. b | H' F0 |
| 482 ;PCIC(0) | | |
| 483 | :H' A00 | PCI_PCISERR |
| 484 | . data. b | H' F0 |
| 485 ;PCIC(1) | | |
| 486 | :H' A20 | PCI_PCIINTA |
| 487 | . data. b | H' F0 |
| 488 ;PCIC(2) | | |
| 489 | :H' A40 | PCI_PCIINTB |
| 490 | . data. b | H' F0 |
| 491 ;PCIC(3) | | |
| 492 | :H' A60 | PCI_PCIINTC |
| 493 | . data. b | H' F0 |
| 494 ;PCIC(4) | | |
| 495 | :H' A80 | PCI_PCIINTD |
| 496 | . data. b | H' F0 |
| 497 ;PCIC(5) | | |
| 498 | :H' AA0 | PCI_PCERR |
| 499 | . data. b | H' F0 |
| 500 | :H' AC0 | PCI_PCIPWD3_1 |
| 501 | . data. b | H' F0 |
| 502 | :H' AE0 | PCI_PCIPWD0 |
| 503 | . data. b | H' F0 |
| 504 ;IRL(7-4) | | |
| 505 | : H' B00-CC0 | IRL74 |
| 506 | . datab. b | 15, H' F0 |
| 507 ;SIOF | | |
| 508 | :H' CE0 | SIOF_SIOFI |
| 509 | . data. b | H' F0 |
| 510 ;MMCIF | | |
| 511 | :H' D00 | MMCIF_FSTAT |
| 512 | . data. b | H' F0 |
| 513 | :H' D20 | MMCIF_TRAN |
| 514 | . data. b | H' F0 |
| 515 | :H' D40 | MMCIF_ERR |

| | | |
|--------------|-----------|--------------|
| 516 | . data. b | H' F0 |
| 517 | :H' D60 | MMCIF_FRDY |
| 518 | . data. b | H' F0 |
| 519 ;DU | | |
| 520 | :H' D80 | MMCIF_FSTAT |
| 521 | . data. b | H' F0 |
| 522 ;GDTA | | |
| 523 | :H' DA0 | GDTA_GACL1 |
| 524 | . data. b | H' F0 |
| 525 | :H' DC0 | GDTA_GAMC1 |
| 526 | . data. b | H' F0 |
| 527 | :H' DE0 | GDTA_GAER1 |
| 528 | . data. b | H' F0 |
| 529 ;TMU-ch3 | | |
| 530 | :H' E00 | TMU_TUNI3 |
| 531 | . data. b | H' F0 |
| 532 ;TMU-ch4 | | |
| 533 | :H' E20 | TMU_TUNI4 |
| 534 | . data. b | H' F0 |
| 535 ;TMU-ch5 | | |
| 536 | :H' E40 | TMU_TUNI5 |
| 537 | . data. b | H' F0 |
| 538 | :H' E60 | Reserved |
| 539 | . data. b | H' F0 |
| 540 ;SSI-ch0 | | |
| 541 | :H' E80 | SSI_SSI10 |
| 542 | . data. b | H' F0 |
| 543 ;SSI-ch1 | | |
| 544 | :H' EA0 | SSI_SSI11 |
| 545 | . data. b | H' F0 |
| 546 ;HAC-ch0 | | |
| 547 | :H' EC0 | HAC_HAC10 |
| 548 | . data. b | H' F0 |
| 549 ;HAC-ch1 | | |
| 550 | :H' EE0 | HAC_HAC11 |
| 551 | . data. b | H' F0 |
| 552 ;FLCTL | | |
| 553 | :H' F00 | FLCTL_FLSTE |
| 554 | . data. b | H' F0 |
| 555 | :H' F20 | FLCTL_FLTEND |
| 556 | . data. b | H' F0 |
| 557 | :H' F40 | FLCTL_FLTRQ0 |
| 558 | . data. b | H' F0 |
| 559 | :H' F60 | FLCTL_FLTRQ1 |
| 560 | . data. b | H' F0 |
| 561 ;GPIO | | |
| 562 | :H' F80 | GPIO_GPIO10 |

```
563      .data.b      H' F0
564      ;H' FA0      GPIO_GPI011
565      .data.b      H' F0
566      ;H' FC0      GPIO_GPI012
567      .data.b      H' F0
568      ;H' FE0      GPIO_GPI013
569      .data.b      H' F0
570      .end
571
572
```

サンプルプログラムリスト” vect.inc”

割込みハンドラに登録する割込みのテーブルです

```

001 ;/*****
002 ;*
003 ;* Device      : SH-4A/SH7785
004 ;*
005 ;* File Name   : vect.inc
006 ;*
007 ;* Abstract    : Definition of Vector.
008 ;*
009 ;* History     : 1.00 (2010-09-30) [Hardware Manual Revision : 1.00]
010 ;*
011 ;* Copyright(c) 2010 Renesas Electronics Corporation.
012 ;*              And Renesas Solutions Corp., All Rights Reserved.
013 ;*
014 ;*****/
015 ;-----
016 ;
017 ; FILE          :vect.inc
018 ; DATE          :Thu, May 13, 2010
019 ; DESCRIPTION   :Definition of Vector
020 ; CPU TYPE      :other
021 ;
022 ; This file is generated by Renesas Project Generator (Ver. 4.16).
023 ;
024 ;-----
025
026 ;<<VECTOR DATA START (POWER ON RESET)>>
027     ;H' 000 Power On Reset (Hitachi-UDI RESET)
028     .global     _PowerON_Reset
029 ;<<VECTOR DATA END (POWER ON RESET)>>
030 ;<<VECTOR DATA START (MANUAL RESET)>>
031     ;H' 020 Manual Reset
032     .global     _Manual_Reset
033 ;<<VECTOR DATA END (MANUAL RESET)>>
034     ;H' 040 Data TLB miss exception(read)
035     .global     _INT_TLB_MISS_READ_EXP
036     ;H' 060 Data TLB miss exception(write)
037     .global     _INT_TLB_MISS_WRITE_EXP
038     ;H' 080 Initial page write exception
039     .global     _INT_TLB_INIT_PAGE_EXP
040     ;H' 0A0 Data TLB protection violation exception (read)
041     .global     _INT_TLB_PROTECT_READ_EXP
042     ;H' 0C0 Data TLB protection violation exception (write)
043     .global     _INT_TLB_PROTECT_WRITE_EXP
044     ;H' 0E0 Data address error (read)
045     .global     _INT_ADR_ERROR_READ

```

| | | | |
|-----|---|--------|--|
| 046 | : | H' 100 | Data address error (write) |
| 047 | . | global | _INT_ADR_ERROR_WRITE |
| 048 | : | H' 120 | FPU exception |
| 049 | . | global | _INT_FPU_EXP |
| 050 | : | H' 140 | Instruction TLB multiple-hit exception |
| 051 | . | global | _TLB_Reset |
| 052 | : | H' 160 | Unconditional trap (TRAPA) |
| 053 | . | global | _INT_TRAP |
| 054 | : | H' 180 | General illegal instruction exception |
| 055 | . | global | _INT_ILLEGAL_INST_EXP |
| 056 | : | H' 1A0 | Slot illegal instruction exception |
| 057 | . | global | _INT_ILLEGAL_SLOT_EXP |
| 058 | : | H' 1C0 | NMI |
| 059 | . | global | _INT_NMI |
| 060 | : | H' 1E0 | USER_BREAK |
| 061 | . | global | _INT_USER_BREAK |
| 062 | | : | H' 200 IRL30_LEVEL15_IRQ7 |
| 063 | . | global | _INT_IRL30_LEVEL15_IRQ7 |
| 064 | | : | H' 220 IRL30_LEVEL14 |
| 065 | . | global | _INT_IRL30_LEVEL14 |
| 066 | | : | H' 240 IRL30_LEVEL13_IRQ0 |
| 067 | . | global | _INT_IRL30_LEVEL13_IRQ0 |
| 068 | | : | H' 260 IRL30_LEVEL12 |
| 069 | . | global | _INT_IRL30_LEVEL12 |
| 070 | | : | H' 280 IRL30_LEVEL11_IRQ1 |
| 071 | . | global | _INT_IRL30_LEVEL11_IRQ1 |
| 072 | | : | H' 2A0 IRL30_LEVEL10 |
| 073 | . | global | _INT_IRL30_LEVEL10 |
| 074 | | : | H' 2C0 IRL30_LEVEL9_IRQ2 |
| 075 | . | global | _INT_IRL30_LEVEL9_IRQ2 |
| 076 | | : | H' 2E0 IRL30_LEVEL8 |
| 077 | . | global | _INT_IRL30_LEVEL8 |
| 078 | | : | H' 300 IRL30_LEVEL7_IRQ3 |
| 079 | . | global | _INT_IRL30_LEVEL7_IRQ3 |
| 080 | | : | H' 320 IRL30_LEVEL6 |
| 081 | . | global | _INT_IRL30_LEVEL6 |
| 082 | | : | H' 340 IRL30_LEVEL5_IRQ4 |
| 083 | . | global | _INT_IRL30_LEVEL5_IRQ4 |
| 084 | | : | H' 360 IRL30_LEVEL4 |
| 085 | . | global | _INT_IRL30_LEVEL4 |
| 086 | | : | H' 380 IRL30_LEVEL3_IRQ5 |
| 087 | . | global | _INT_IRL30_LEVEL3_IRQ5 |
| 088 | | : | H' 3A0 IRL30_LEVEL2 |
| 089 | . | global | _INT_IRL30_LEVEL2 |
| 090 | | : | H' 3C0 IRL30_LEVEL1_IRQ6 |
| 091 | . | global | _INT_IRL30_LEVEL1_IRQ6 |
| 092 | | : | H' 3E0 Reserved |

| | | |
|--------------|----------|------------------|
| 093 | . global | _INT_Reserved3E0 |
| 094 | :H' 400 | Reserved |
| 095 | . global | _INT_Reserved400 |
| 096 | :H' 420 | Reserved |
| 097 | . global | _INT_Reserved420 |
| 098 | :H' 440 | Reserved |
| 099 | . global | _INT_Reserved440 |
| 100 | :H' 460 | Reserved |
| 101 | . global | _INT_Reserved460 |
| 102 | :H' 480 | Reserved |
| 103 | . global | _INT_Reserved480 |
| 104 | :H' 4A0 | Reserved |
| 105 | . global | _INT_Reserved4A0 |
| 106 | :H' 4C0 | Reserved |
| 107 | . global | _INT_Reserved4C0 |
| 108 | :H' 4E0 | Reserved |
| 109 | . global | _INT_Reserved4E0 |
| 110 | :H' 500 | Reserved |
| 111 | . global | _INT_Reserved500 |
| 112 | :H' 520 | Reserved |
| 113 | . global | _INT_Reserved520 |
| 114 | :H' 540 | Reserved |
| 115 | . global | _INT_Reserved540 |
| 116 ;WDT | | |
| 117 | :H' 560 | WDT_ITI |
| 118 | . global | _INT_WDT_ITI |
| 119 ;TMU-ch0 | | |
| 120 | :H' 580 | TMU_TUNI0 |
| 121 | . global | _INT_TMU_TUNI0 |
| 122 ;TMU-ch1 | | |
| 123 | :H' 5A0 | TMU_TUNI1 |
| 124 | . global | _INT_TMU_TUNI1 |
| 125 ;TMU-ch2 | | |
| 126 | :H' 5C0 | TMU_TUNI2 |
| 127 | . global | _INT_TMU_TUNI2 |
| 128 | :H' 5E0 | TMU_TICPI2 |
| 129 | . global | _INT_TMU_TICPI2 |
| 130 ;H-UDI | | |
| 131 | :H' 600 | H-UDI1 |
| 132 | . global | _INT_H_UDI1 |
| 133 ;DMAC(0) | | |
| 134 | :H' 620 | DMINT0 |
| 135 | . global | _INT_DMAC_DMINT0 |
| 136 | :H' 640 | DMINT1 |
| 137 | . global | _INT_DMAC_DMINT1 |
| 138 | :H' 660 | DMINT2 |
| 139 | . global | _INT_DMAC_DMINT2 |

| | | |
|-----|----------|-------------------|
| 140 | :H' 680 | DMINT3 |
| 141 | .global | _INT_DMAC_DMINT3 |
| 142 | :H' 6A0 | DMINT4 |
| 143 | .global | _INT_DMAC_DMINT4 |
| 144 | :H' 6C0 | DMINT5 |
| 145 | .global | _INT_DMAC_DMINT5 |
| 146 | :H' 6E0 | DMAEO (ch0-5) |
| 147 | .global | _INT_DMAC_DMAEO |
| 148 | :SCIF(0) | |
| 149 | :H' 700 | SCIF_ERI0 |
| 150 | .global | _INT_SCIF_ERI0 |
| 151 | :H' 720 | SCIF_RXI0 |
| 152 | .global | _INT_SCIF_RXI0 |
| 153 | :H' 740 | SCIF_BRI0 |
| 154 | .global | _INT_SCIF_BRI0 |
| 155 | :H' 760 | SCIF_TXI0 |
| 156 | .global | _INT_SCIF_TXI0 |
| 157 | :SCIF(1) | |
| 158 | :H' 780 | SCIF_ERI1 |
| 159 | .global | _INT_SCIF_ERI1 |
| 160 | :H' 7A0 | SCIF_RXI1 |
| 161 | .global | _INT_SCIF_RXI1 |
| 162 | :H' 7C0 | SCIF_BRI1 |
| 163 | .global | _INT_SCIF_BRI1 |
| 164 | :H' 7E0 | SCIF_TXI1 |
| 165 | .global | _INT_SCIF_TXI1 |
| 166 | :H' 800 | Reserved |
| 167 | .global | _INT_Reserved800 |
| 168 | :H' 820 | Reserved |
| 169 | .global | _INT_Reserved820 |
| 170 | :H' 840 | Reserved |
| 171 | .global | _INT_Reserved840 |
| 172 | :H' 860 | Reserved |
| 173 | .global | _INT_Reserved860 |
| 174 | :DMAC(1) | |
| 175 | :H' 880 | DMINT6 |
| 176 | .global | _INT_DMAC_DMINT6 |
| 177 | :H' 8A0 | DMINT7 |
| 178 | .global | _INT_DMAC_DMINT7 |
| 179 | :H' 8C0 | DMINT8 |
| 180 | .global | _INT_DMAC_DMINT8 |
| 181 | :H' 8E0 | DMINT9 |
| 182 | .global | _INT_DMAC_DMINT9 |
| 183 | :H' 900 | DMINT10 |
| 184 | .global | _INT_DMAC_DMINT10 |
| 185 | :H' 920 | DMINT11 |
| 186 | .global | _INT_DMAC_DMINT11 |

| | | |
|---------------|----------|--------------------|
| 187 | ;H' 940 | DMAE1 (ch6-11) |
| 188 | .global | _INT_DMAE1 |
| 189 ;HSP1 | | |
| 190 | ;H' 960 | HSP1_SPI1 |
| 191 | .global | _INT_HSP1_SPI1 |
| 192 ;SCIF(2) | | |
| 193 | ;H' 980 | SCIF_SCIF12 |
| 194 | .global | _INT_SCIF_SCIF12 |
| 195 ;SCIF(3) | | |
| 196 | ;H' 9A0 | SCIF_SCIF13 |
| 197 | .global | _INT_SCIF_SCIF13 |
| 198 ;SCIF(4) | | |
| 199 | ;H' 9C0 | SCIF_SCIF14 |
| 200 | .global | _INT_SCIF_SCIF14 |
| 201 ;SCIF(5) | | |
| 202 | ;H' 9E0 | SCIF_SCIF15 |
| 203 | .global | _INT_SCIF_SCIF15 |
| 204 ;PCIC(0) | | |
| 205 | ;H' A00 | PCI_PCISERR |
| 206 | .global | _INT_PCI_PCISERR |
| 207 ;PCIC(1) | | |
| 208 | ;H' A20 | PCI_PCINTA |
| 209 | .global | _INT_PCI_PCINTA |
| 210 ;PCIC(2) | | |
| 211 | ;H' A40 | PCI_PCINTB |
| 212 | .global | _INT_PCI_PCINTB |
| 213 ;PCIC(3) | | |
| 214 | ;H' A60 | PCI_PCINTC |
| 215 | .global | _INT_PCI_PCINTC |
| 216 ;PCIC(4) | | |
| 217 | ;H' A80 | PCI_PCINTD |
| 218 | .global | _INT_PCI_PCINTD |
| 219 ;PCIC(5) | | |
| 220 | ;H' AA0 | PCI_PCERR |
| 221 | .global | _INT_PCI_PCERR |
| 222 | ;H' AC0 | PCI_PCIPWD3_1 |
| 223 | .global | _INT_PCI_PCIPWD3_1 |
| 224 | ;H' AE0 | PCI_PCIPWD0 |
| 225 | .global | _INT_PCI_PCIPWD0 |
| 226 ;IRL(7-4) | | |
| 227 | ; H' B00 | IRL74_LEVEL15 |
| 228 | .global | _INT_IRL74_LEVEL15 |
| 229 | ; H' B20 | IRL74_LEVEL14 |
| 230 | .global | _INT_IRL74_LEVEL14 |
| 231 | ; H' B40 | IRL74_LEVEL13 |
| 232 | .global | _INT_IRL74_LEVEL13 |
| 233 | ; H' B60 | IRL74_LEVEL12 |

| | | |
|-----|----------|--------------------|
| 234 | .global | _INT_IRL74_LEVEL12 |
| 235 | ; H' B80 | IRL74_LEVEL11 |
| 236 | .global | _INT_IRL74_LEVEL11 |
| 237 | ; H' BA0 | IRL74_LEVEL10 |
| 238 | .global | _INT_IRL74_LEVEL10 |
| 239 | ; H' BC0 | IRL74_LEVEL9 |
| 240 | .global | _INT_IRL74_LEVEL9 |
| 241 | ; H' BE0 | IRL74_LEVEL8 |
| 242 | .global | _INT_IRL74_LEVEL8 |
| 243 | ; H' C00 | IRL74_LEVEL7 |
| 244 | .global | _INT_IRL74_LEVEL7 |
| 245 | ; H' C20 | IRL74_LEVEL6 |
| 246 | .global | _INT_IRL74_LEVEL6 |
| 247 | ; H' C40 | IRL74_LEVEL5 |
| 248 | .global | _INT_IRL74_LEVEL5 |
| 249 | ; H' C60 | IRL74_LEVEL4 |
| 250 | .global | _INT_IRL74_LEVEL4 |
| 251 | ; H' C80 | IRL74_LEVEL3 |
| 252 | .global | _INT_IRL74_LEVEL3 |
| 253 | ; H' CA0 | IRL74_LEVEL2 |
| 254 | .global | _INT_IRL74_LEVEL2 |
| 255 | ; H' CC0 | IRL74_LEVEL1 |
| 256 | .global | _INT_IRL74_LEVEL1 |
| 257 | :SIOF | |
| 258 | ;H' CE0 | SIOF_SIOF I |
| 259 | .global | _INT_SIOF_SIOF I |
| 260 | :MMCIF | |
| 261 | ;H' D00 | MMCIF_FSTAT |
| 262 | .global | _INT_MMCIF_FSTAT |
| 263 | ;H' D20 | MMCIF_TRAN |
| 264 | .global | _INT_MMCIF_TRAN |
| 265 | ;H' D40 | MMCIF_ERR |
| 266 | .global | _INT_MMCIF_ERR |
| 267 | ;H' D60 | MMCIF_FRDY |
| 268 | .global | _INT_MMCIF_FRDY |
| 269 | :DU | |
| 270 | ;H' D80 | DU_DUI |
| 271 | .global | _INT_DU_DUI |
| 272 | :GDTA | |
| 273 | ;H' DA0 | GDTA_GACL I |
| 274 | .global | _INT_GDTA_GACL I |
| 275 | ;H' DC0 | GDTA_GAMC I |
| 276 | .global | _INT_GDTA_GAMC I |
| 277 | ;H' DE0 | GDTA_GAER I |
| 278 | .global | _INT_GDTA_GAER I |
| 279 | :TMU-ch3 | |
| 280 | ;H' E00 | TMU_TUNI3 |

| | | |
|-----|------------------------------------|-------------------|
| 281 | . global | _INT_TMU_TUN13 |
| 282 | ;TMU-ch4 | |
| 283 | ;H' E20 | TMU_TUN14 |
| 284 | . global | _INT_TMU_TUN14 |
| 285 | ;TMU-ch5 | |
| 286 | ;H' E40 | TMU_TUN15 |
| 287 | . global | _INT_TMU_TUN15 |
| 288 | ;H' E60 | Reserved |
| 289 | . global | _INT_ReservedE60 |
| 290 | ;SSI-ch0 | |
| 291 | ;H' E80 | SSI_SSI10 |
| 292 | . global | _INT_SSI_SSI10 |
| 293 | ;SSI-ch1 | |
| 294 | ;H' EA0 | SSI_SSI11 |
| 295 | . global | _INT_SSI_SSI11 |
| 296 | ;HAC-ch0 | |
| 297 | ;H' EC0 | HAC_HAC10 |
| 298 | . global | _INT_HAC_HAC10 |
| 299 | ;HAC-ch1 | |
| 300 | ;H' EE0 | HAC_HAC11 |
| 301 | . global | _INT_HAC_HAC11 |
| 302 | ;FLCTL | |
| 303 | ;H' F00 | FLCTL_FLSTE |
| 304 | . global | _INT_FLCTL_FLSTE |
| 305 | ;H' F20 | FLCTL_FLTEND |
| 306 | . global | _INT_FLCTL_FLTEND |
| 307 | ;H' F40 | FLCTL_FLTRQ0 |
| 308 | . global | _INT_FLCTL_FLTRQ0 |
| 309 | ;H' F60 | FLCTL_FLTRQ1 |
| 310 | . global | _INT_FLCTL_FLTRQ1 |
| 311 | ;GPIO | |
| 312 | ;H' F80 | GPIO_GPIO10 |
| 313 | . global | _INT_GPIO_GPIO10 |
| 314 | ;H' FA0 | GPIO_GPIO11 |
| 315 | . global | _INT_GPIO_GPIO11 |
| 316 | ;H' FC0 | GPIO_GPIO12 |
| 317 | . global | _INT_GPIO_GPIO12 |
| 318 | ;H' FE0 | GPIO_GPIO13 |
| 319 | . global | _INT_GPIO_GPIO13 |
| 320 | | |
| 321 | | |
| 322 | | |
| 323 | ;<<VECTOR DATA START (TBL RESET)>> | |
| 324 | ;H' 140 TBL Reset (DATA TBL Reset) | |
| 325 | . global | _TBL_Reset |
| 326 | ;<<VECTOR DATA END (TBL RESET)>> | |
| 327 | | |

サンプルプログラムリスト” intprg.c”

割込み処理関数を割込みハンドラに登録しています。

```
001 /*****
002 * DISCLAIMER
003
004 * This software is supplied by Renesas Electronics Corporation. and is only
005 * intended for use with Renesas products. No other uses are authorized.
006
007 * This software is owned by Renesas Electronics Corporation. and is protected under
008 * all applicable laws, including copyright laws.
009
010 * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
014 * DISCLAIMED.
015
016 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 * ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021
022 * Renesas reserves the right, without notice, to make changes to this
023 * software and to discontinue the availability of this software.
024 * By using this software, you agree to the additional terms and
025 * conditions found by accessing the following link:
026 * http://www.renesas.com/disclaimer
027 *****/
028 /* Copyright (C) 2010. Renesas Electronics Corporation., All Rights Reserved.*/
029 /*"FILE COMMENT"***** Technical reference data *****
030 * System Name : SH7785 Sample Program
031 * File Name : intprg.c
032 * Abstract : SH7785 初期設定例 Sample Program
033 * Version : Ver 1.00
034 * Device : SH7785
035 * Tool-Chain : High-performance Embedded Workshop (Version 4.07.00.007)
036 * : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 * OS : None
038 * H/W Platform : ルネサスエレクトロニクス製 SH-4A ボード 型番 ROP7785LC0011RL
039 * Description : SH7785 初期設定例のサンプルプログラムです。
040 * : intprg.src を C 言語形式に変更した。
041 * :
042 * Operation :
043 * Limitation :
044 * :
045 *****/
```

```
046 * History      : 30.SEP.2010 Ver. 1.00 First Release
047 *"FILE COMMENT END"*****
048 #include <machine.h>
049 #include "iodefine.h"
050
051 /* --- RAM allocation variable declaration --- */
052
053 #pragma section IntPRG
054 /* H'040 Data TLB miss exception(read) */
055 void INT_TLB_MISS_READ_EXP(void)
056 {
057 }
058
059 /* H'060 Data TLB miss exception(write) */
060 void INT_TLB_MISS_WRITE_EXP(void)
061 {
062 }
063
064 /* H'080 Initial page write exception */
065 void INT_TLB_INIT_PAGE_EXP(void)
066 {
067 }
068
069 /* H'0A0 Data TLB protection violation exception (read) */
070 void INT_TLB_PROTECT_READ_EXP(void)
071 {
072 }
073
074 /* H'0C0 Data TLB protection violation exception (write) */
075 void INT_TLB_PROTECT_WRITE_EXP(void)
076 {
077 }
078
079 /* H'0E0 Data address error (read) */
080 void INT_ADR_ERROR_READ(void)
081 {
082 }
083
084 /* H'100 Data address error (write) */
085 void INT_ADR_ERROR_WRITE(void)
086 {
087 }
088
089 /* H'120 FPU exception */
090 void INT_FPU_EXP(void)
091 {
092 }
```

```
093
094 /* H' 140 Instruction TLB multiple-hit exception */
095 void TLB_Reset(void)
096 {
097 }
098
099 /* H' 160 Unconditional trap(TRAPA) */
100 void INT_TRAP(void)
101 {
102 }
103
104 /* H' 180 General illegal instruction exception */
105 void INT_ILLEGAL_INST_EXP(void)
106 {
107 }
108
109 /* H' 1A0 Slot illegal instruction exception */
110 void INT_ILLEGAL_SLOT_EXP(void)
111 {
112 }
113
114 /* H' 1C0 NMI */
115 void INT_NMI(void)
116 {
117 }
118
119 /* H' 1E0 USER_BREAK */
120 void INT_USER_BREAK(void)
121 {
122 }
123
124 /* H' 200 IRL30_LEVEL15_IRQ7 */
125 void INT_IRL30_LEVEL15_IRQ7(void)
126 {
127 }
128
129 /* H' 220 IRL30_LEVEL14 */
130 void INT_IRL30_LEVEL14(void)
131 {
132 }
133
134 /* H' 240 IRL30_LEVEL13_IRQ0 */
135 void INT_IRL30_LEVEL13_IRQ0(void)
136 {
137 }
138
139 /* H' 260 IRL30_LEVEL12 */
```

```
140 void INT_IRL30_LEVEL12(void)
141 {
142 }
143
144 /* H' 280 IRL30_LEVEL11_IRQ1 */
145 void INT_IRL30_LEVEL11_IRQ1(void)
146 {
147 }
148
149 /* H' 2A0 IRL30_LEVEL10 */
150 void INT_IRL30_LEVEL10(void)
151 {
152 }
153
154 /* H' 2C0 IRL30_LEVEL9_IRQ2 */
155 void INT_IRL30_LEVEL9_IRQ2(void)
156 {
157 }
158
159 /* H' 2E0 IRL30_LEVEL8 */
160 void INT_IRL30_LEVEL8(void)
161 {
162 }
163
164 /* H' 300 IRL30_LEVEL7_IRQ3 */
165 void INT_IRL30_LEVEL7_IRQ3(void)
166 {
167 }
168
169 /* H' 320 IRL30_LEVEL6 */
170 void INT_IRL30_LEVEL6(void)
171 {
172 }
173
174 /* H' 340 IRL30_LEVEL5_IRQ4 */
175 void INT_IRL30_LEVEL5_IRQ4(void)
176 {
177 }
178
179 /* H' 360 IRL30_LEVEL4 */
180 void INT_IRL30_LEVEL4(void)
181 {
182 }
183
184 /* H' 380 IRL30_LEVEL3_IRQ5 */
185 void INT_IRL30_LEVEL3_IRQ5(void)
186 {
```



```
187 }
188
189 /* H' 3A0 IRL30_LEVEL2 */
190 void INT_IRL30_LEVEL2(void)
191 {
192 }
193
194 /* H' 3C0 IRL30_LEVEL1_IRQ6 */
195 void INT_IRL30_LEVEL1_IRQ6(void)
196 {
197 }
198
199 /* H' 560 WDT overflow interrupt */
200 void INT_WDT_ITI(void)
201 {
202 }
203
204 /* H' 580 TMU ch-0 underflow interrupt */
205 void INT_TMU_TUNIO(void)
206 {
207 }
208
209 /* H' 5A0 TMU ch-1 underflow interrupt */
210 void INT_TMU_TUNI1(void)
211 {
212 }
213
214 /* H' 5C0 TMU ch-2 underflow interrupt */
215 void INT_TMU_TUNI2(void)
216 {
217 }
218
219 /* H' 5E0 TMU ch-2 input capture interrupt */
220 void INT_TMU_TICPI2(void)
221 {
222 }
223
224 /* H' 600 H-UDI Update interrupt */
225 void INT_H_UDI1(void)
226 {
227 }
228
229 /* H' 620 DMAC0 interrupt */
230 void INT_DMAC_DMINT0(void)
231 {
232   irq_disable( _DMAC0 );
233   dmac0_irq();
```

```
234 }
235
236 /* H' 640 DMAC1 interrupt */
237 void INT_DMADMINT1(void)
238 {
239 }
240
241 /* H' 660 DMAC2 interrupt */
242 void INT_DMADMINT2(void)
243 {
244 }
245
246 /* H' 680 DMAC3 interrupt */
247 void INT_DMADMINT3(void)
248 {
249 }
250
251 /* H' 6A0 DMAC4 interrupt */
252 void INT_DMADMINT4(void)
253 {
254 }
255
256 /* H' 6C0 DMAC5 interrupt */
257 void INT_DMADMINT5(void)
258 {
259 }
260
261 /* H' 6E0 DMAC 0-5 Address error interrupt */
262 void INT_DMADMAEO(void)
263 {
264 }
265
266 /* H' 700 SCIF0 error interrupt */
267 void INT_SCIF_ERI0(void)
268 {
269 }
270
271 /* H' 720 SCIF0 Recive interrupt */
272 void INT_SCIF_RXI0(void)
273 {
274 }
275
276 /* H' 740 SCIF0 Brake interrupt */
277 void INT_SCIF_BRI0(void)
278 {
279 }
280
```

```
281 /* H' 760 SCIF0 Trans interrupt */
282 void INT_SCIF_TXI0(void)
283 {
284 }
285
286 /* H' 780 SCIF1 error interrupt */
287 void INT_SCIF_ERI1(void)
288 {
289 }
290
291 /* H' 7A0 SCIF1 Recive interrupt */
292 void INT_SCIF_RXI1(void)
293 {
294 }
295
296 /* H' 7C0 SCIF1 Brake interrupt */
297 void INT_SCIF_BRI1(void)
298 {
299 }
300
301 /* H' 7E0 SCIF1 Trans interrupt */
302 void INT_SCIF_TXI1(void)
303 {
304 }
305
306 /* H' 880 DMAC6 interrupt */
307 void INT_DMAC_DMINT6(void)
308 {
309 }
310
311 /* H' 8A0 DMAC7 interrupt */
312 void INT_DMAC_DMINT7(void)
313 {
314 }
315
316 /* H' 8C0 DMAC8 interrupt */
317 void INT_DMAC_DMINT8(void)
318 {
319 }
320
321 /* H' 8E0 DMAC9 interrupt */
322 void INT_DMAC_DMINT9(void)
323 {
324 }
325
326 /* H' 900 DMAC11 interrupt */
327 void INT_DMAC_DMINT10(void)
```

```
328 {
329 }
330
331 /* H'920 DMAC11 interrupt */
332 void INT_DMAC_DMINT11(void)
333 {
334 }
335
336 /* H'940 DMAC 6-11 Address error interrupt */
337 void INT_DMAC_DMACE1(void)
338 {
339 }
340
341 /* H'960 HSPI interrupt */
342 void INT_HSPI_SPI1(void)
343 {
344 }
345
346 /* H'980 SCIF2 interrupt */
347 void INT_SCIF_SCIF12(void)
348 {
349 }
350
351 /* H'9A0 SCIF3 interrupt */
352 void INT_SCIF_SCIF13(void)
353 {
354 }
355
356 /* H'9C0 SCIF4 interrupt */
357 void INT_SCIF_SCIF14(void)
358 {
359 }
360
361 /* H'9E0 SCIF5 interrupt */
362 void INT_SCIF_SCIF15(void)
363 {
364 }
365
366 /* H'A00 PCI SERR interrupt */
367 void INT_PCI_PCISERR(void)
368 {
369 }
370
371 /* H'A20 PCI INTA interrupt */
372 void INT_PCI_PCIINTA(void)
373 {
374 }
```

```
375
376 /* H' A40 PCI INTB interrupt */
377 void INT_PCI_PCIINTB(void)
378 {
379 }
380
381 /* H' A60 PCI INTC interrupt */
382 void INT_PCI_PCIINTC(void)
383 {
384 }
385
386 /* H' A80 PCI INTD interrupt */
387 void INT_PCI_PCIINTD(void)
388 {
389 }
390
391 /* H' AA0 PCI ERR interrupt */
392 void INT_PCI_PCERR(void)
393 {
394 }
395
396 /* H' AC0 PCI PWD3-1 interrupt */
397 void INT_PCI_PCIPWD3_1(void)
398 {
399 }
400
401 /* H' AE0 PCI PWD0 interrupt */
402 void INT_PCI_PCIPWD0(void)
403 {
404 }
405
406 /* H' B00 IRL74_LEVEL15 interrupt */
407 void INT_IRL74_LEVEL15(void)
408 {
409 }
410
411 /* H' B20 IRL74_LEVEL14 interrupt */
412 void INT_IRL74_LEVEL14(void)
413 {
414 }
415
416 /* H' B40 IRL74_LEVEL13 interrupt */
417 void INT_IRL74_LEVEL13(void)
418 {
419 }
420
421 /* H' B60 IRL74_LEVEL12 interrupt */
```

```
422 void INT_IRL74_LEVEL12(void)
423 {
424 }
425
426 /* H'B80 IRL74_LEVEL11 interrupt */
427 void INT_IRL74_LEVEL11(void)
428 {
429 }
430
431 /* H'BA0 IRL74_LEVEL10 interrupt */
432 void INT_IRL74_LEVEL10(void)
433 {
434 }
435
436 /* H'BC0 IRL74_LEVEL9 interrupt */
437 void INT_IRL74_LEVEL9(void)
438 {
439 }
440
441 /* H'BE0 IRL74_LEVEL8 interrupt */
442 void INT_IRL74_LEVEL8(void)
443 {
444 }
445
446 /* H'CO0 IRL74_LEVEL7 interrupt */
447 void INT_IRL74_LEVEL7(void)
448 {
449 }
450
451 /* H'C20 IRL74_LEVEL6 interrupt */
452 void INT_IRL74_LEVEL6(void)
453 {
454 }
455
456 /* H'C40 IRL74_LEVEL5 interrupt */
457 void INT_IRL74_LEVEL5(void)
458 {
459 }
460
461 /* H'C60 IRL74_LEVEL4 interrupt */
462 void INT_IRL74_LEVEL4(void)
463 {
464 }
465
466 /* H'C80 IRL74_LEVEL3 interrupt */
467 void INT_IRL74_LEVEL3(void)
468 {
```

```
469 }
470
471 /* H' CA0 IRL74_LEVEL2 interrupt */
472 void INT_IRL74_LEVEL2(void)
473 {
474 }
475
476 /* H' CC0 IRL74_LEVEL1 interrupt */
477 void INT_IRL74_LEVEL1(void)
478 {
479 }
480
481 /* H' CE0 SIOF interrupt */
482 void INT_SIOF_SIOFI(void)
483 {
484 }
485
486 /* H' D00 MMCIF FIFO Empty/Full interrupt */
487 void INT_MMCIF_FSTAT(void)
488 {
489 }
490
491 /* H' D20 MMCIF Transfer end interrupt */
492 void INT_MMCIF_TRAN(void)
493 {
494 }
495
496 /* H' D40 MMCIF Error interrupt */
497 void INT_MMCIF_ERR(void)
498 {
499 }
500
501 /* H' D60 MMCIF FIFO Ready interrupt */
502 void INT_MMCIF_FRDY(void)
503 {
504 }
505
506 /* H' D80 DU interrupt */
507 void INT_DU_DUI(void)
508 {
509 }
510
511 /* H' DA0 GDTA CL end interrupt */
512 void INT_GDTA_GACLI(void)
513 {
514 }
515
```

```
516 /* H'DC0 GDTA MC end interrupt */
517 void INT_GDTA_GAMC1(void)
518 {
519 }
520
521 /* H'DE0 GDTA Error interrupt */
522 void INT_GDTA_GAER1(void)
523 {
524 }
525
526 /* H'E00 TMU ch-3 underflow interrupt */
527 void INT_TMU_TUNI3(void)
528 {
529 }
530
531 /* H'E20 TMU ch-4 underflow interrupt */
532 void INT_TMU_TUNI4(void)
533 {
534 }
535
536 /* H'E40 TMU ch-5 underflow interrupt */
537 void INT_TMU_TUNI5(void)
538 {
539 }
540
541 /* H'E80 SSI ch-0 interrupt */
542 void INT_SSI_SSI10(void)
543 {
544 }
545
546 /* H'EA0 SSI ch-1 interrupt */
547 void INT_SSI_SSI11(void)
548 {
549 }
550
551 /* H'EC0 HAC ch-0 interrupt */
552 void INT_HAC_HAC10(void)
553 {
554 }
555
556 /* H'EE0 HAC ch-1 interrupt */
557 void INT_HAC_HAC11(void)
558 {
559 }
560
561 /* H'F00 FLCTL error interrupt */
562 void INT_FLCTL_FLSTE(void)
```



```
563 {
564 }
565
566 /* H'F20 FLCTL Transfer end interrupt */
567 void INT_FLCTL_FLTEND(void)
568 {
569 }
570
571 /* H'F40 FLCTL FIFO0 request interrupt */
572 void INT_FLCTL_FLTRQ0(void)
573 {
574 }
575
576 /* H'F60 FLCTL FIFO1 request interrupt */
577 void INT_FLCTL_FLTRQ1(void)
578 {
579 }
580
581 /* H'F80 GPIO0 interrupt */
582 void INT_GPIO_GPII0I0(void)
583 {
584 }
585
586 /* H'FA0 GPIO1 interrupt */
587 void INT_GPIO_GPII0I1(void)
588 {
589 }
590
591 /* H'FC0 GPIO2 interrupt */
592 void INT_GPIO_GPII0I2(void)
593 {
594 }
595
596 /* H'FE0 GPIO3 interrupt */
597 void INT_GPIO_GPII0I3(void)
598 {
599 }
600
601 void INT_Reserved(void)
602 {
603 }
```

サンプルプログラムリスト” lowlevelinit src”

LBSC 及び DBSC2 の初期設定を行っています。

```
001 ;/*****
002 ;* DISCLAIMER
003 ;
004 ;* This software is supplied by Renesas Electronics Corporation. and is only
005 ;* intended for use with Renesas products. No other uses are authorized.
006 ;
007 ;* This software is owned by Renesas Electronics Corporation. and is protected under
008 ;* all applicable laws, including copyright laws.
009 ;
010 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
014 ;* DISCLAIMED.
015 ;
016 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021 ;
022 ;* Renesas reserves the right, without notice, to make changes to this
023 ;* software and to discontinue the availability of this software.
024 ;* By using this software, you agree to the additional terms and
025 ;* conditions found by accessing the following link:
026 ;* http://www.renesas.com/disclaimer
027 ;*****/
028 ;/* Copyright (C) 2010. Renesas Electronics Corporation., All Rights Reserved.*/
029 ;/*"FILE COMMENT"***** Technical reference data *****
030 ;* System Name : SH7785 Sample Program
031 ;* File Name : lowlevelinit.src
032 ;* Abstract : SH7785 初期設定例 Sample Program
033 ;* Version : Ver 1.00
034 ;* Device : SH7785
035 ;* Tool-Chain : High-performance Embedded Workshop (Version 4.04.01.001)
036 ;* : C/C++ Compiler Package for SuperH Family (V.9.02release00)
037 ;* OS : None
038 ;* H/W Platform : ルネサスエレクトロニクス製 SH-4A ボード 型番 ROP7785LC0011RL
039 ;* Description : SH7785 初期設定例のサンプルプログラムです。
040 ;* :
041 ;* :
042 ;* Operation :
043 ;* Limitation :
044 ;* :
045 ;*****/
```

```
046 ;* History      : 30.SEP.2010 Ver. 1.00 First Release
047 ;*""FILE COMMENT END""*****/
048 ;-----
049 ;
050 ; FILE          :lowlevelinit.src
051 ; DATE          :Thu, May 13, 2010
052 ; DESCRIPTION  :DBSC2/LBSC Initial
053 ; CPU TYPE     :Other
054 ;
055 ; This file is generated by Renesas Project Generator (Ver.4.16).
056 ;
057 ;-----
058
059 .macro write32 addr, data
060 mov.l ¥addr, r1
061 mov.l ¥data, r0
062 mov.l r0, @r1
063 .endm
064
065 .macro write16 addr, data
066 mov.l ¥addr, r1
067 mov.l ¥data, r0
068 mov.w r0, @r1
069 .endm
070
071 .macro write8 addr, data
072 mov.l ¥addr, r1
073 mov.l ¥data, r0
074 mov.b r0, @r1
075 .endm
076
077 .macro wait_timer time
078 mov.l ¥time, r3
079 loop¥@:
080 nop
081 tst r3, r3
082 bf/s loop¥@
083 dt r3
084 .endm
085
086 .include "lowlevelinit.inc"
087
088 .export _lowlevel_init
089
090 _lowlevel_init:
091
092 ; /*----- LBSC -----*/
```

```
093  write32 #MMSEL_R_A, #MMSEL_R_D
094
095 ; /*----- DBSC2 -----*/
096  write32 #DBSC2_DBCONF_A, #DBSC2_DBCONF_D
097  write32 #DBSC2_DBTR0_A, #DBSC2_DBTR0_D
098  write32 #DBSC2_DBTR1_A, #DBSC2_DBTR1_D
099  write32 #DBSC2_DBTR2_A, #DBSC2_DBTR2_D
100  write32 #DBSC2_DBFREQ_A, #DBSC2_DBFREQ_D1
101  write32 #DBSC2_DBFREQ_A, #DBSC2_DBFREQ_D2
102  wait_timer #WAIT_200US
103
104  write32 #DBSC2_DBDICODTOCD_A, #DBSC2_DBDICODTOCD_D
105  write32 #DBSC2_DBCMDCNT_A, #DBSC2_DBCMDCNT_D_CKE_H
106  wait_timer #3333
107  write32 #DBSC2_DBCMDCNT_A, #DBSC2_DBCMDCNT_D_PALL
108  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_EMRS2
109  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_EMRS3
110  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_EMRS1_1
111  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_MRS_1
112  write32 #DBSC2_DBCMDCNT_A, #DBSC2_DBCMDCNT_D_PALL
113  write32 #DBSC2_DBCMDCNT_A, #DBSC2_DBCMDCNT_D_REF
114  write32 #DBSC2_DBCMDCNT_A, #DBSC2_DBCMDCNT_D_REF
115  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_MRS_2
116  wait_timer #WAIT_200US
117
118  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_EMRS1_2
119  write32 #DBSC2_DBMRCNT_A, #DBSC2_DBMRCNT_D_EMRS1_1
120
121  write32 #DBSC2_DBEN_A, #DBSC2_DBEN_D
122  write32 #DBSC2_DBRFCNT1_A, #DBSC2_DBRFCNT1_D
123  write32 #DBSC2_DBRFCNT2_A, #DBSC2_DBRFCNT2_D
124  write32 #DBSC2_DBRFCNT0_A, #DBSC2_DBRFCNT0_D
125
126 ; /*----- LBSC -----*/
127  write32 #BCR_A, #BCR_D
128
129 ; mov.l #PASCR_A, r0
130 ; mov.l @r0, r2
131 ; mov.l #PASCR_32BIT_MODE, r1
132 ; tst r1, r2
133 ; bt lpsc_29bit
134
135 .aifdef CONFIG_CS0
136  write32 #CS0_BCR_A, #CS0_BCR_D
137  write32 #CS0_WCR_A, #CS0_WCR_D
138 .aendi
139
```

```
140 .aifdef CONFIG_CS1
141 write32 #CS1_BCR_A, #CS1_BCR_D
142 write32 #CS1_WCR_A, #CS1_WCR_D
143 .aendi
144
145 .aifdef CONFIG_CS2
146 write32 #CS2_BCR_A, #CS2_BCR_D
147 write32 #CS2_WCR_A, #CS2_WCR_D
148 .aendi
149
150 .aifdef CONFIG_CS3
151 write32 #CS3_BCR_A, #CS3_BCR_D
152 write32 #CS3_WCR_A, #CS3_WCR_D
153 .aendi
154
155 .aifdef CONFIG_CS4
156 write32 #CS4_BCR_A, #CS4_BCR_D
157 write32 #CS4_WCR_A, #CS4_WCR_D
158 .aendi
159
160 .aifdef CONFIG_CS5
161 write32 #CS5_BCR_A, #CS5_BCR_D
162 write32 #CS5_WCR_A, #CS5_WCR_D
163 .aendi
164
165 .aifdef CONFIG_CS6
166 write32 #CS6_BCR_A, #CS6_BCR_D
167 write32 #CS6_WCR_A, #CS6_WCR_D
168 .aendi
169
170 bra lpsc_end
171 nop
172
173 lpsc_end:
174 .aifdef CONFIG_SH_32BIT
175 ; /*----- set PMB -----*/
176 write32 #PASCR_A, #PASCR_29BIT_D
177 write32 #MMUCR_A, #MMUCR_D
178
179 ; /*****
180 ; * ent virt phys v sz c wt
181 ; * 0 0xa0000000 0x00000000 1 64M 0 0
182 ; * 1 0xa4000000 0x04000000 1 16M 0 0
183 ; * 2 0xa6000000 0x08000000 1 16M 0 0
184 ; * 9 0x88000000 0x48000000 1 128M 1 1
185 ; * 10 0x90000000 0x50000000 1 128M 1 1
186 ; * 11 0x98000000 0x58000000 1 128M 1 1
```

```
187 ; * 13 0xa8000000 0x48000000 1 128M 0 0
188 ; * 14 0xb0000000 0x50000000 1 128M 0 0
189 ; * 15 0xb8000000 0x58000000 1 128M 0 0
190
191 write32 #PMB_ADDR_FLASH_A, #PMB_ADDR_FLASH_D
192 write32 #PMB_DATA_FLASH_A, #PMB_DATA_FLASH_D
193 write32 #PMB_ADDR_CPLD_A, #PMB_ADDR_CPLD_D
194 write32 #PMB_DATA_CPLD_A, #PMB_DATA_CPLD_D
195 write32 #PMB_ADDR_USB_A, #PMB_ADDR_USB_D
196 write32 #PMB_DATA_USB_A, #PMB_DATA_USB_D
197 write32 #PMB_ADDR_DDR_C1_A, #PMB_ADDR_DDR_C1_D
198 write32 #PMB_DATA_DDR_C1_A, #PMB_DATA_DDR_C1_D
199 write32 #PMB_ADDR_DDR_C2_A, #PMB_ADDR_DDR_C2_D
200 write32 #PMB_DATA_DDR_C2_A, #PMB_DATA_DDR_C2_D
201 write32 #PMB_ADDR_DDR_C3_A, #PMB_ADDR_DDR_C3_D
202 write32 #PMB_DATA_DDR_C3_A, #PMB_DATA_DDR_C3_D
203 write32 #PMB_ADDR_DDR_N1_A, #PMB_ADDR_DDR_N1_D
204 write32 #PMB_DATA_DDR_N1_A, #PMB_DATA_DDR_N1_D
205 write32 #PMB_ADDR_DDR_N2_A, #PMB_ADDR_DDR_N2_D
206 write32 #PMB_DATA_DDR_N2_A, #PMB_DATA_DDR_N2_D
207 write32 #PMB_ADDR_DDR_N3_A, #PMB_ADDR_DDR_N3_D
208 write32 #PMB_DATA_DDR_N3_A, #PMB_DATA_DDR_N3_D
209
210 write32 #PASCRA, #PASCRA_INIT
211 mov.l #DUMMY_ADDR, r0
212 icbi @r0
213 .aendi
214
215 write32 #CCR_A, #CCR_D
216 rts
217 nop
218
219
220
221 .end
```

サンプルプログラムリスト” lowlevelinit.inc”

LBSC 及び DBSC2 の初期設定例です。

```

001
002 ; The choice of the area to initialize.
003 ;-----
004 ;CONFIG_SH_32BIT: .define "" ;Address 32Bit Mode
005
006
007 CONFIG_CS0:      .define "" ;Initial CS0
008 CONFIG_CS1:      .define "" ;Initial CS1
009 ;
010 .aifdef CONFIG_SH_32BIT
011 CONFIG_CS2:      .define "" ;Initial CS2
012 CONFIG_CS3:      .define "" ;Initial CS3
013 .aendi
014
015 CONFIG_CS4:      .define "" ;Initial CS4
016 CONFIG_CS5:      .define "" ;Initial CS5
017 CONFIG_CS6:      .define "" ;Initial CS6
018 ;CONFIG_CS5_PCMCIA: .define ""
019 ;CONFIG_CS6_PCMCIA: .define ""
020
021 ;-----
022
023 ;/*----- DBSC2 -----*/
024 DBSC2_BASE:      .equ      H' fe800000
025 DBSC2_DBSTATE_A: .equ      DBSC2_BASE + H' 0c
026 DBSC2_DBEN_A:    .equ      DBSC2_BASE + H' 10
027 DBSC2_DBCMDCNT_A: .equ      DBSC2_BASE + H' 14
028 DBSC2_DBCONF_A:  .equ      DBSC2_BASE + H' 20
029 DBSC2_DBTRO_A:   .equ      DBSC2_BASE + H' 30
030 DBSC2_DBTR1_A:   .equ      DBSC2_BASE + H' 34
031 DBSC2_DBTR2_A:   .equ      DBSC2_BASE + H' 38
032 DBSC2_DBRFCNT0_A: .equ      DBSC2_BASE + H' 40
033 DBSC2_DBRFCNT1_A: .equ      DBSC2_BASE + H' 44
034 DBSC2_DBRFCNT2_A: .equ      DBSC2_BASE + H' 48
035 DBSC2_DBRFSTS_A: .equ      DBSC2_BASE + H' 4c
036 DBSC2_DBFREQ_A:  .equ      DBSC2_BASE + H' 50
037 DBSC2_DBDICODTOCD_A: .equ      DBSC2_BASE + H' 54
038 DBSC2_DBMRCNT_A: .equ      DBSC2_BASE + H' 60
039 DDR_DUMMY_ACCESS_A: .equ      H' 40000000
040
041 DBSC2_DBCONF_D:  .equ      H' 00630002
042 DBSC2_DBTRO_D:   .equ      H' 050d2604
043 DBSC2_DBTR1_D:   .equ      H' 00040204
044 DBSC2_DBTR2_D:   .equ      H' 02120308
045 DBSC2_DBFREQ_D1: .equ      H' 00000000

```

```

046 DBSC2_DBFREQ_D2:                .equ    H' 0000100
047 DBSC2_DBDI CODTOCD_D:          .equ    H' 000f0907
048
049 DBSC2_DBMRCNT_D_EMRS2:         .equ    H' 00020000
050 DBSC2_DBMRCNT_D_EMRS3:         .equ    H' 00030000
051 DBSC2_DBMRCNT_D_EMRS1_1:      .equ    H' 00010006
052 DBSC2_DBMRCNT_D_EMRS1_2:      .equ    H' 00010386
053 DBSC2_DBMRCNT_D_MRS_1:        .equ    H' 00000952
054 DBSC2_DBMRCNT_D_MRS_2:        .equ    H' 00000852
055
056 DBSC2_DBPCNT0_D3:              .equ    H' 00000080
057 DBSC2_DBRFCNT1_D:             .equ    H' 00000926
058 DBSC2_DBRFCNT2_D:             .equ    H' 00fe00fe
059 DBSC2_DBRFCNT0_D:             .equ    H' 00010000
060
061 DBSC2_DBCMDCNT_D_CKE_H:        .equ    H' 00000003
062 DBSC2_DBCMDCNT_D_PALL:        .equ    H' 00000002
063 DBSC2_DBCMDCNT_D_REF:         .equ    H' 00000004
064
065 DBSC2_DBEN_D:                  .equ    H' 00000001
066
067 WAIT_200US:                    .equ    33333
068
069 ;/*----- LBSC -----*/
070 PASCR_A:                        .equ    H' ff000070
071 PASCR_32BIT_MODE:              .equ    H' 80000000 /* check booting mode */
072
073 MMSELR_A:                       .equ    H' fc400020
074 BCR_A:                          .equ    H' FF801000
075 CS0_BCR_A:                      .equ    H' FF802000
076 CS1_BCR_A:                      .equ    H' FF802010
077 CS2_BCR_A:                      .equ    H' FF802020
078 CS3_BCR_A:                      .equ    H' FF802030
079 CS4_BCR_A:                      .equ    H' FF802040
080 CS5_BCR_A:                      .equ    H' FF802050
081 CS6_BCR_A:                      .equ    H' FF802060
082 CS0_WCR_A:                      .equ    H' FF802008
083 CS1_WCR_A:                      .equ    H' FF802018
084 CS2_WCR_A:                      .equ    H' FF802028
085 CS3_WCR_A:                      .equ    H' FF802038
086 CS4_WCR_A:                      .equ    H' FF802048
087 CS5_WCR_A:                      .equ    H' FF802058
088 CS6_WCR_A:                      .equ    H' FF802068
089 CS5_PCMCIA_A:                  .equ    H' FF802070
090 CS6_PCMCIA_A:                  .equ    H' FF802080
091
092 .aifdef CONFIG_SH_32BIT

```



```

093 MMSELR_D:                .equ    H' a5a50005
094 .aelse
095 MMSELR_D:                .equ    H' a5a50002
096 .aendi
097
098 BCR_D:                   .equ    H' 80000003
099 CS0_BCR_D:               .equ    H' 22222340
100 CS0_WCR_D:               .equ    H' 00111118
101 CS1_BCR_D:               .equ    H' 11111100
102 CS1_WCR_D:               .equ    H' 33333303
103 CS4_BCR_D:               .equ    H' 11111300
104 CS4_WCR_D:               .equ    H' 00101012
105
106 ;/* USB setting : 32bit mode = CS2, 29bit mode = CS5 */
107 .aifdef CONFIG_SH_32BIT
108 CS2_BCR_D:                .equ    H' 11111200
109 CS2_WCR_D:                .equ    H' 00020004
110 .aelse
111 CS5_BCR_D:                .equ    H' 11111200
112 CS5_WCR_D:                .equ    H' 00020004
113 .aendi
114
115 ;/* SD setting : 32bit mode = CS3, 29bit mode = CS6 */
116 .aifdef CONFIG_SH_32BIT
117 CS3_BCR_D:                .equ    H' 00000300
118 CS3_WCR_D:                .equ    H' 00030108
119 .aelse
120 CS6_BCR_D:                .equ    H' 00000300
121 CS6_WCR_D:                .equ    H' 00030108
122 .aendi
123
124 ;/* I2C setting : 32bit mode = CS5, 29bit mode = CS1(already setting) */
125 .aifdef CONFIG_SH_32BIT
126 CS5_BCR_D:                .equ    H' 11111100
127 CS5_WCR_D:                .equ    H' 00000003
128 .aendi
129
130 ;/* Chache */
131 CCR_A:                     .equ    H' ff00001c
132 CCR_D:                     .equ    H' 0000090b
133
134 ;PMB
135 .aifdef CONFIG_SH_32BIT
136 PMB
137 PMB_ADDR_ARRAY:           .equ    H' f6100000
138 PMB_ADDR_ENTRY:           .equ    8
139 PMB_VPN:                   .equ    24

```

```

140
141 PMB_DATA_ARRAY:                .equ   H' f7100000
142 PMB_DATA_ENTRY:                .equ   8
143 PMB_PPN:                       .equ   24
144 PMB_UB:                         .equ   9      /* Buffered write */
145 PMB_V:                          .equ   8      /* Valid */
146 PMB_SZ1:                       .equ   7      /* Page size (upper bit) */
147 PMB_SZ0:                       .equ   4      /* Page size (lower bit) */
148 PMB_C:                          .equ   3      /* Cacheability */
149 PMB_WT:                         .equ   0      /* Write-through */
150
151 #define PMB_ADDR_BASE(entry)     (PMB_ADDR_ARRAY + (entry << PMB_ADDR_ENTRY))
152 #define PMB_DATA_BASE(entry)    (PMB_DATA_ARRAY + (entry << PMB_DATA_ENTRY))
153 #define mk_pmb_addr_val(vpn)    ((vpn << PMB_VPN))
154 #define mk_pmb_data_val(ppn, ub, v, sz1, sz0, c, wt)  ¥
155 ;      ((ppn << PMB_PPN) | (ub << PMB_UB) | ¥
156 ;      (v << PMB_V) | (sz1 << PMB_SZ1) | ¥
157 ;      (sz0 << PMB_SZ0) | (c << PMB_C) | ¥
158 ;      (wt << PMB_WT))
159
160 PMB_ADDR_FLASH_A:               .equ   PMB_ADDR_ARRAY + 0
161 PMB_ADDR_CPLD_A:               .equ   PMB_ADDR_ARRAY + 1 << PMB_ADDR_ENTRY
162 PMB_ADDR_USB_A:                .equ   PMB_ADDR_ARRAY + 2 << PMB_ADDR_ENTRY
163 PMB_ADDR_DDR_C1_A:             .equ   PMB_ADDR_ARRAY + 9 << PMB_ADDR_ENTRY
164 PMB_ADDR_DDR_C2_A:             .equ   PMB_ADDR_ARRAY + 10 << PMB_ADDR_ENTRY
165 PMB_ADDR_DDR_C3_A:            .equ   PMB_ADDR_ARRAY + 11 << PMB_ADDR_ENTRY
166 PMB_ADDR_DDR_N1_A:            .equ   PMB_ADDR_ARRAY + 13 << PMB_ADDR_ENTRY
167 PMB_ADDR_DDR_N2_A:            .equ   PMB_ADDR_ARRAY + 14 << PMB_ADDR_ENTRY
168 PMB_ADDR_DDR_N3_A:            .equ   PMB_ADDR_ARRAY + 15 << PMB_ADDR_ENTRY
169
170 PMB_ADDR_FLASH_D:              .equ   H' a0 << PMB_VPN
171 PMB_ADDR_CPLD_D:               .equ   H' a4 << PMB_VPN
172 PMB_ADDR_USB_D:                .equ   H' a6 << PMB_VPN
173 PMB_ADDR_DDR_C1_D:             .equ   H' 88 << PMB_VPN
174 PMB_ADDR_DDR_C2_D:             .equ   H' 90 << PMB_VPN
175 PMB_ADDR_DDR_C3_D:            .equ   H' 98 << PMB_VPN
176 PMB_ADDR_DDR_N1_D:            .equ   H' a8 << PMB_VPN
177 PMB_ADDR_DDR_N2_D:            .equ   H' b0 << PMB_VPN
178 PMB_ADDR_DDR_N3_D:            .equ   H' b8 << PMB_VPN
179
180 PMB_DATA_FLASH_A:              .equ   PMB_DATA_ARRAY + 0
181 PMB_DATA_CPLD_A:               .equ   PMB_DATA_ARRAY + 1 << PMB_DATA_ENTRY
182 PMB_DATA_USB_A:                .equ   PMB_DATA_ARRAY + 2 << PMB_DATA_ENTRY
183 PMB_DATA_DDR_C1_A:            .equ   PMB_DATA_ARRAY + 9 << PMB_DATA_ENTRY
184 PMB_DATA_DDR_C2_A:            .equ   PMB_DATA_ARRAY + 10 << PMB_DATA_ENTRY
185 PMB_DATA_DDR_C3_A:            .equ   PMB_DATA_ARRAY + 11 << PMB_DATA_ENTRY
186 PMB_DATA_DDR_N1_A:            .equ   PMB_DATA_ARRAY + 13 << PMB_DATA_ENTRY

```

```

187 PMB_DATA_DDR_N2_A:                .equ PMB_DATA_ARRAY + 14 << PMB_DATA_ENTRY
188 PMB_DATA_DDR_N3_A:                .equ PMB_DATA_ARRAY + 15 << PMB_DATA_ENTRY
189
190 ;/*          ppn  ub v s1 s0  c  wt */
191 PMB_DATA_FLASH_D:                  .equ H' 00 << PMB_PPN | 1 << PMB_UB | 1 << PMB_V | 0 << PMB_SZ1 | 1 <<
PMB_SZ0 | 0 << PMB_C | 1 << PMB_WT
192 PMB_DATA_CPLD_D:                   .equ H' 04 << PMB_PPN | 1 << PMB_UB | 1 << PMB_V | 0 << PMB_SZ1 | 0 <<
PMB_SZ0 | 0 << PMB_C | 1 << PMB_WT
193 PMB_DATA_USB_D:                    .equ H' 08 << PMB_PPN | 1 << PMB_UB | 1 << PMB_V | 0 << PMB_SZ1
| 0 << PMB_SZ0 | 0 << PMB_C | 1 << PMB_WT
194 PMB_DATA_DDR_C1_D:                  .equ H' 48 << PMB_PPN | 0 << PMB_UB | 1 << PMB_V | 1 << PMB_SZ1
| 0 << PMB_SZ0 | 1 << PMB_C | 1 << PMB_WT
195 PMB_DATA_DDR_C2_D:                  .equ H' 50 << PMB_PPN | 0 << PMB_UB | 1 << PMB_V | 1 << PMB_SZ1
| 0 << PMB_SZ0 | 1 << PMB_C | 1 << PMB_WT
196 PMB_DATA_DDR_C3_D:                  .equ H' 58 << PMB_PPN | 0 << PMB_UB | 1 << PMB_V | 1 << PMB_SZ1
| 0 << PMB_SZ0 | 1 << PMB_C | 1 << PMB_WT
197 PMB_DATA_DDR_N1_D:                  .equ H' 48 << PMB_PPN | 1 << PMB_UB | 1 << PMB_V | 1 << PMB_SZ1
| 0 << PMB_SZ0 | 0 << PMB_C | 1 << PMB_WT
198 PMB_DATA_DDR_N2_D:                  .equ H' 50 << PMB_PPN | 1 << PMB_UB | 1 << PMB_V | 1 << PMB_SZ1
| 0 << PMB_SZ0 | 0 << PMB_C | 1 << PMB_WT
199 PMB_DATA_DDR_N3_D:                  .equ H' 58 << PMB_PPN | 1 << PMB_UB | 1 << PMB_V | 1 << PMB_SZ1
| 0 << PMB_SZ0 | 0 << PMB_C | 1 << PMB_WT
200
201 DUMMY_ADDR:                         .equ H' a0000000
202 PASC_R_29BIT_D:                     .equ H' 00000000
203 PASC_R_INIT:                         .equ H' 80000080  /* check booting mode */
204 MMUCR_A:                             .equ H' ff000010
205 MMUCR_D:                             .equ H' 00000004  /* clear ITLB */
206
207 .aendi

```

6. 参考ドキュメント

- ソフトウェアマニュアル
SH4-A ソフトウェアマニュアル(RJJ09B0090)
(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
SH7785 グループハードウェアマニュアル(RJJ09B0285)
(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。

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。



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/contact/>