
SH7785 グループ

R01AN0243JJ0101

Rev.1.01

2010.12.01

SH7785 PCIC 初期設定例

要旨

この資料は、SH7785 の PCIC 初期設定に必要な設定例を示します。

動作確認デバイス

SH7785

目次

| | |
|-------------------|----|
| 1. はじめに..... | 2 |
| 2. PCICの動作説明..... | 5 |
| 3. 応用例の説明..... | 12 |
| 4. 参考プログラム例..... | 25 |
| 5. 実行結果..... | 44 |
| 6. 参考ドキュメント..... | 44 |

1. はじめに

1.1 仕様

本アプリケーションノートでは、2章でPCICのハードウェアマニュアルを補足する説明を行い、3章ではPCIC初期設定後にシリアルコンソールにPCIデバイスのデバイスID等の表示を行う応用例を掲載していません。

1.2 使用機能

- ピンファンクションコントローラ(PFC)
- シリアルコミュニケーションインタフェース(SCIF1)
- PCIコントローラ(PCIC)

1.3 適用条件

| | |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 評価ボード: | ルネサスエレクトロニクス製 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 (32bit モード時 : 512M バイト) Elpida 製 EDE1108ACSE-6E-E (4 個) |
| PCI デバイス (ch0): | 1Gbit-Ether コントローラ - Realtek 製 RTL8169 |
| (ch1): | 2ch-SATA コントローラ - SiliconImage 製 SI3512 |
| (ch2): | 拡張コネクタ |
| (ch3): | 拡張コネクタ |
| 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

1.4 本アプリケーションノートで使用する用語の説明

- コンフィグレーションレジスタ空間
PCI コンプライアントデバイスのプログラミングモデルおよび使用ルールを定義します。
- IO 空間
PCI ローカルバス仕様で定義されているローカルベース空間です。
- メモリ空間
PCI ローカルバス仕様で定義されているメモリベース空間です。

1.5 本アプリケーションノートの適用範囲

本アプリケーションノートは、OS 非搭載で PCIC の初期設定後に PCI の各デバイスの VendorID および DeviceID の値をシリアルコンソールに表示する。また、Ether デバイスの MAC アドレスをシリアルコンソールに表示する。PCIC の基本的な使用方法について説明します。以下の機能については、本アプリケーションノートの説明対象外となります。

- PCI ターゲット機能
- PCI パワーマネジメント
- ノーマルモード(外部バスアービタモード)
- 擬似ラウンドロビンまたは固定優先順位アービトレーション
- パリティチェックおよびエラーレポート
- 排他的アクセス
- 4 種類の外部割り込み(INTA, INTB, INTC, INTD)

1.6 関連アプリケーションノート

本資料の参考プログラムは、「SH7785 グループアプリケーションノート SH7785 初期設定例 (R01AN0242JJ0101)」で動作確認しています。併せて参照ください。

2. PCICの動作説明

2.1 PCICの概要

PCIC は PCI デバイスにアクセスすることができるプロトコルおよびインタフェースを備えたモジュールです。

本アプリケーションノートでは、初期設定後に接続されている各 PCI デバイスの VendorID および DeviceID と Ether コントローラの MAC アドレスをシリアルコンソールに表示する設定例を紹介します。

図 1 に本アプリケーションノートでの PCI デバイスの構成図を示します。

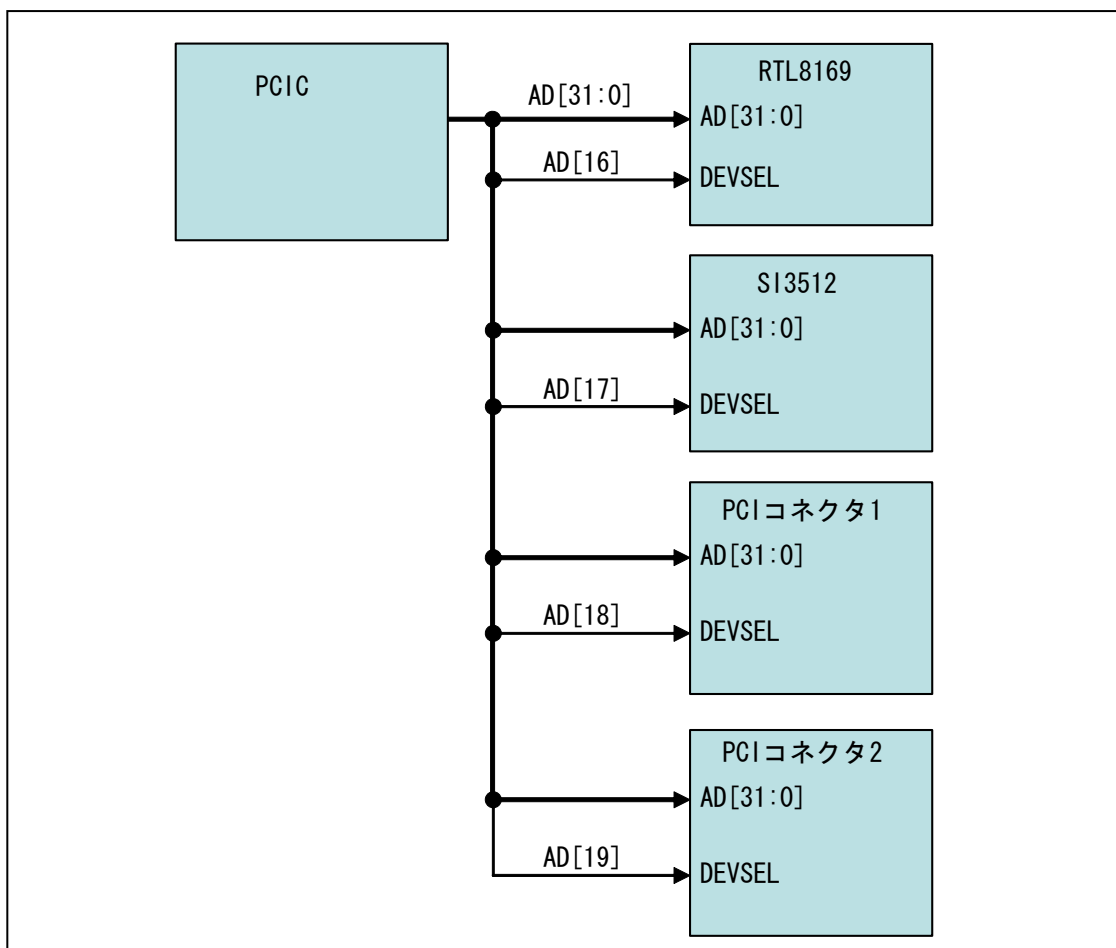


図 1 PCI デバイス構成図

図 2 に PCI インタフェースの基本タイミングチャートを示します。

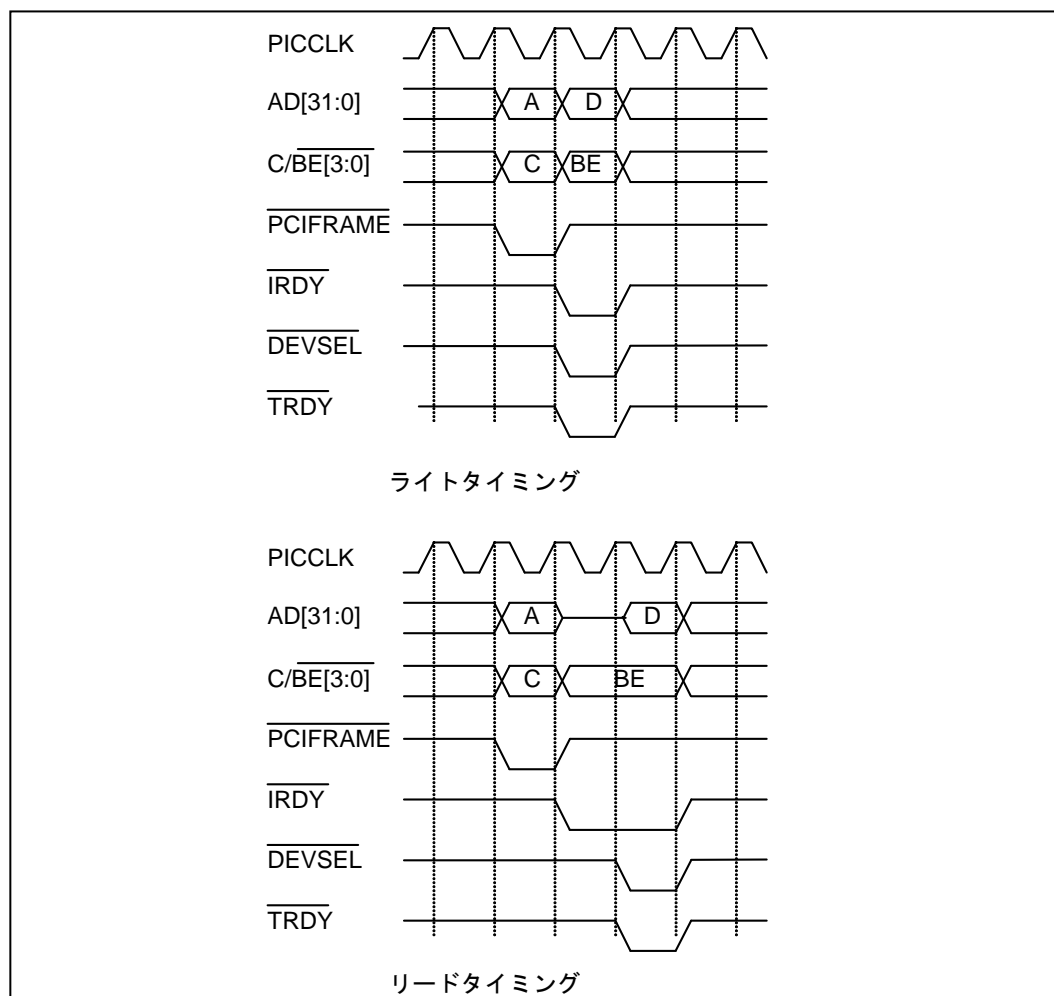


図 2 PCI 基本タイミングチャート

2.1.1 サポートしているPCIコマンド

表1にPCICがサポートしているコマンド一覧を示します。

表 1 PCIコマンド一覧

| C/BE[3:0]信号 | コマンド |
|-------------|---------------|
| 0001 | スペシャルサイクル |
| 0010 | I/O リード |
| 0011 | I/O ライト |
| 0110 | メモリリード |
| 0111 | メモリライト |
| 1010 | コンフィグレーションリード |
| 1011 | コンフィグレーションライト |

2.1.2 PCICメモリマップ

表2にPCICのメモリマップを示します。

本アプリケーションノートでは29ビットアドレスモード,PCIメモリ空間は0のみを使用しています。

表 2 PCICメモリマップ

| メモリ空間 | 29ビットアドレスモード時 物理アドレス | 物理アドレスサイズ |
|----------------------------|-------------------------|-----------|
| PCIメモリ空間1 (エリア4:PCI選択時) | H'10000000~H'13FFFFFF | 64Mバイト |
| PCIメモリ空間0 | H'FD000000~H'FDFFFFFF | 16Mバイト |
| コントロールレジスタ空間 | H'FE000000~FE03FFFF | 256Kバイト |
| PCI内部レジスタ | H'FE040000~H'FE07FFFF | 256Kバイト |
| リザーブ | H'FE080000~H'FE1FFFFFF | 1.5Mバイト |
| PCI I/O 空間 | H'FE200000~H'FE3FFFFFF | 2Mバイト |

PCICには4種類のアドレス空間があります。メモリ空間,コントロールレジスタ空間,PCI内部レジスタ空間,I/O空間です。

2.1.3 PCIメモリ空間へのアクセス

図3に SuperHyway バスから PCI バスへのメモリマップを示します。

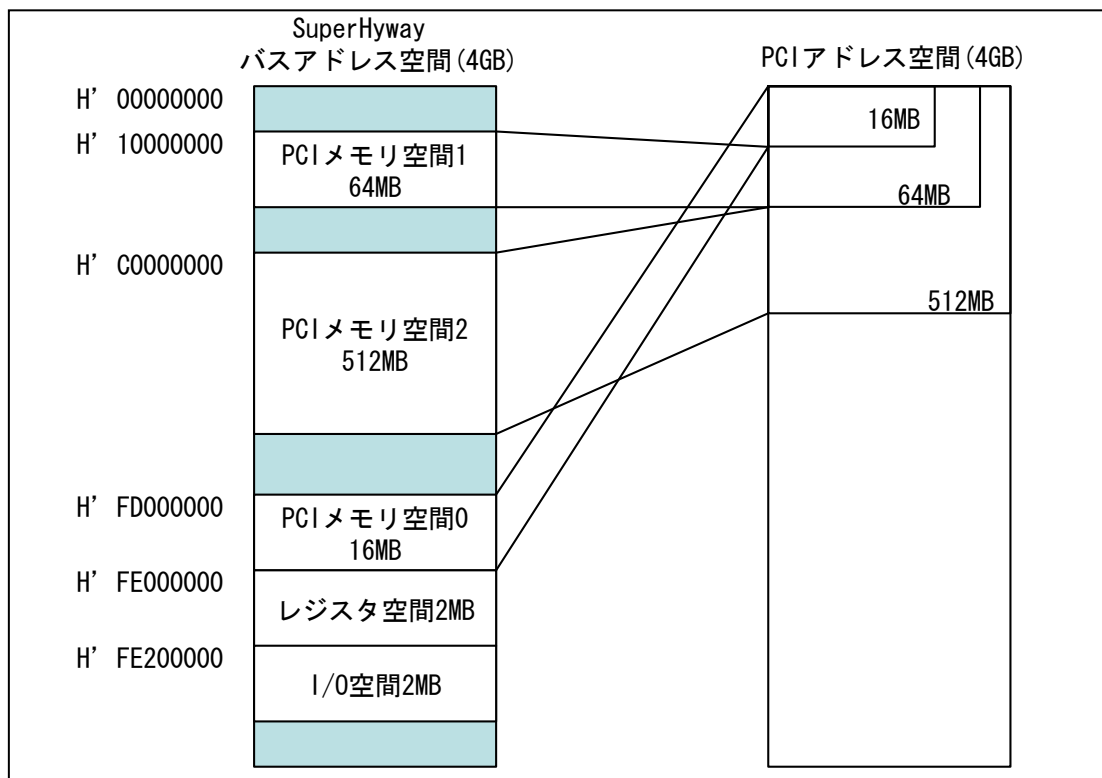


図 3 PCIメモリ空間アクセス

(1)PCI メモリ空間へのアクセス

PCI メモリ空間にアクセスするには、PCI メモリバンクレジスタ(PCIMBR)および PCI メモリバンクマスクレジスタ(PCIMBMR)を設定します。

PCI メモリ空間 0 の場合、中間の 6 ビット([23:18])は、PCI メモリバンクマスクレジスタ(PCIMBMR0)によって制御されます。本アプリケーションノートは PCI メモリ空間 0 のみを使用している為、PCI 空間メモリ空間 0 以外の説明は致しません。

- PCIMBMR0 [23:18] B'1111 11 の場合 : PCI バスアドレス[23:18] = SuperHyway バスアドレス[23:18]
- PCIMBMR0 [23:18] B'0000 00 の場合 : PCI バスアドレス[23:18] = PCIMBR0[23:18]

SuperHyway バス上のアドレス上位 8 ビット[31:24]は、PCI メモリバンクレジスタ PCIMBR0[31:24]に置き換えられます。

図 4 に Superhyway バスから PCI メモリ空間へのアクセスを示します。

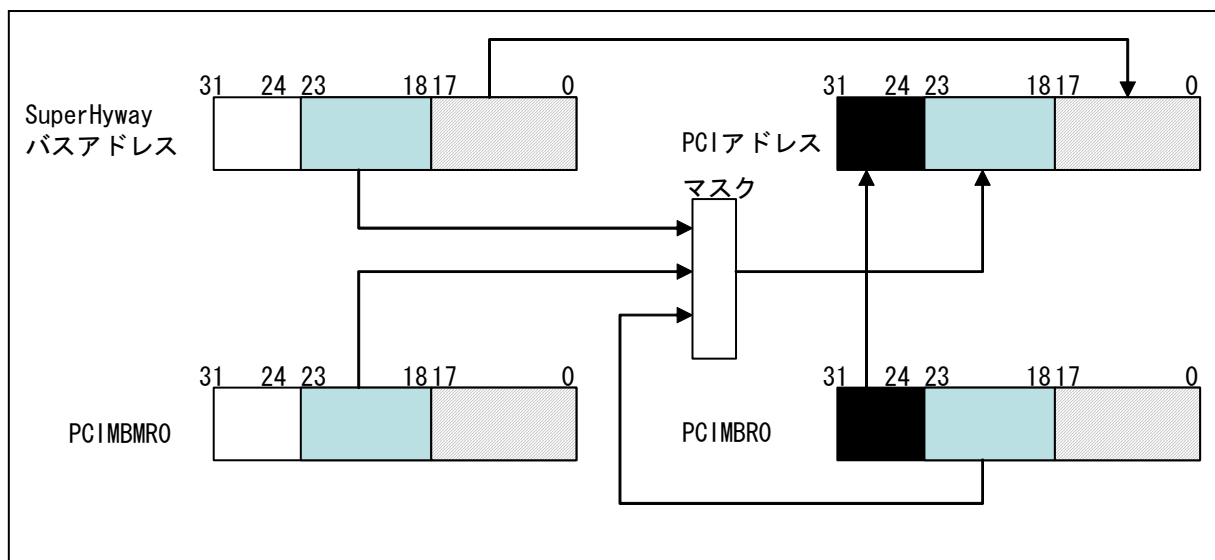


図 4 SuperHyway バスから PCI メモリ空間へのアクセス

- バースト転送

PCI メモリ空間アクセスにおいて、CPU または DMA から 32 バイトバーストのアクセスを連続で実行すると、PCI バス上で 32 バイト以上(すなわち 64, 96 バイトなど)のバースト転送を行います。

(2)PCII/O 空間へのアクセス

PCII/O 空間へアクセスするには、PCIIO バンクレジスタ(PCIIOBR)および PCIIO バンクマスクレジスタ(PCIIOBMR)を設定します。

SuperHyway バス上のアドレス下位 15 ビット ([17:3]) は、そのまま PCI バス上に送られます。

- PCIIOBMR[20:18] B'111 の場合 : PCI バスアドレス[20:18] = SuperHyway バスアドレス[20:18]
- PCIIOBMR[20:18] B'000 の場合 : PCI バスアドレス[20:18] = PCIIOBR[20:18]

SuperHyway バス上のアドレス上位 11 ビット[31:21]は、PCI I/O バンクレジスタ PCIIOBR[31:21]に置き換えられます。

図 5 に SuperHyway バスから PCII/O 空間へのアクセスを示します。

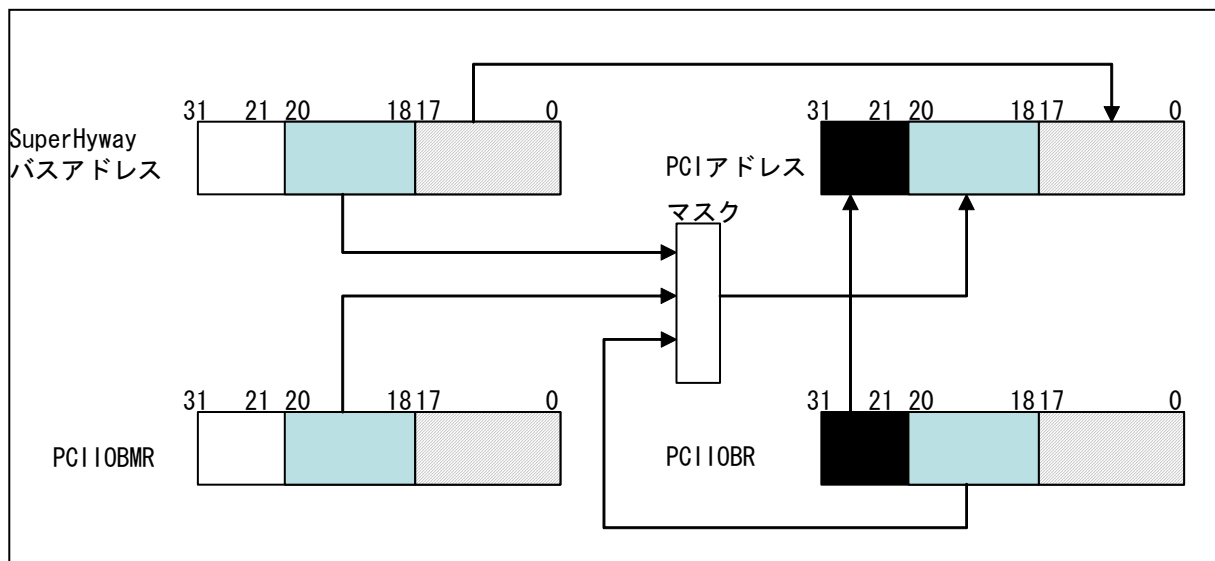


図 5 SuperHyway バスから PCII/O 空間へのアクセス

(3)PCIC 内部レジスタアクセス

全ての PCIC 内部レジスタ(PCIC コントロールレジスタ、PCI コンフィグレーションレジスタおよび PCI ローカルレジスタ)は、CPU からアクセス可能です。アクセスサイズとしては 4 バイト、2 バイトおよび 1 バイトサイズの転送をサポートします。

(4)PCI コンフィグレーション空間へのアクセス

PCIPIO アドレスレジスタ(PCIPAR)と PCIPIO データレジスタ(PCIPDR)を設定してコンフィグレーションサイクルを発行してコンフィグレーション空間へアクセスします。

PCIPAR をセットして、PCIPDR をリードおよびライトをするとコンフィグレーションサイクルが発行されます。

PCIPAR のデバイス番号を 0 にセットすると、AD16 が '1' になりその他は '0' になります。

PCIPAR のデバイス番号を 1 にセットすると、AD17 が '1' になりその他は '0' になります。

2.2 PCICの仕様

PCIC の主な仕様を表 3 に示します。

表 3 PCIC 仕様一覧

| 項目 | 仕様 |
|-------------|-----------------------------------------------------------|
| PCI レビジョン | PCI2.2 のサブセットサポート |
| バス動作周波数 | 33MHz または 66MHz |
| バス幅 | 32 ビット |
| ホスト/ターゲット機能 | ホスト、ターゲット機能ともにサポート 本アプリケーションノートではターゲット機能は取り扱いません。 |
| 割込み | 4 つの割込み(INTA, INTB, INTC, INTD) 本アプリケーションノートでは取り扱いません。 |
| バースト転送 | 最大 32 バイト転送可能 |

3. 応用例の説明

PCIC を動作させる為の参考例として、PCI デバイスの概要と設定例を説明します。

3.1 シリアルコンソールの仕様

表 4 にシリアルコンソールの仕様を示します。

表 4 シリアルコンソール仕様

| 項目 | 仕様 |
|-----------|-----------|
| ボーレート | 115200bps |
| データ | 8 ビット |
| パリティビット | 無し |
| ストップビット | 1 ビット |
| フローコントロール | 無し |

3.2 RTL8169 の概要

表 5 に PCIC に接続している RTL8169 の概要を示します。

表 5 RTL8169 の概要

| 項目 | 内容 |
|----------------|--------------------------------|
| デバイス | 1GBit Ether コントローラデバイス |
| メーカー | Realtek |
| PCIC からのデバイス番号 | 0(AD16 が接続) |
| VendorID | 0x10EC |
| DeviceID | 0x8169 |
| MAC アドレス | R0P7785LC0011RL ボード上に印字されています。 |

3.3 SI3512 の概要

表 6 に PCIC に接続している SI3512 の概要を示します。

表 6 SI3512 の概要

| 項目 | 内容 |
|----------------|---------------------|
| デバイス | SATA-2ch コントローラデバイス |
| メーカー | SiliconImage |
| PCIC からのデバイス番号 | 1(AD17 が接続) |
| VendorID | 0x1039 |
| DeviceID | 0x3512 |

3.4 参考プログラムの仕様

ここでは参考プログラムの仕様と各処理のフローチャートを説明します。

3.4.1 仕様

(1)PCIC を初期設定をします。

(2)PCI デバイス番号 0(RTL8169)の VendorID と DeviceID をシリアルコンソールに表示します。

(3) PCI デバイス番号 1(SI3512) の VendorID と DeviceID をシリアルコンソールに表示します。

(4)RTL8169 の MAC アドレスをシリアルコンソールに表示します。

(5)SI3512 のメモリ空間に Write/Read を行い、Write データ、Read データをコンソールに表示します。

(6) SI3512 のメモリ空間に DMA で 32 バイト×4=128 バイトのバースト転送を実行

(SI3512 が 1 データフェーズごとに STOP 信号をアサートしディスコネクトする為、バースト転送にならない)

3.4.2 参考プログラムメインフロー

図 6~8 に参考プログラムのメインフローを示します。

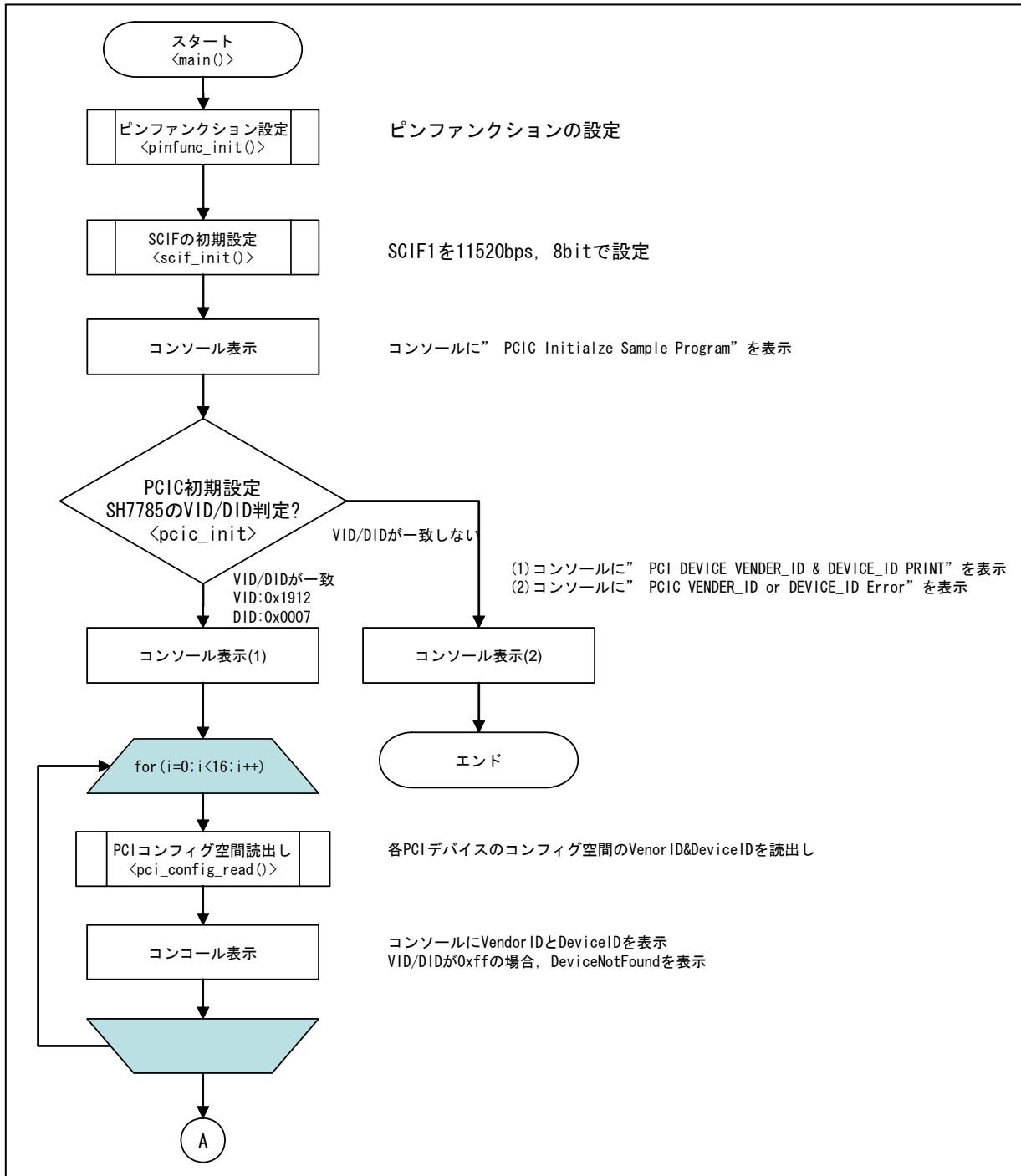


図 6 参考プログラムメインフロー1

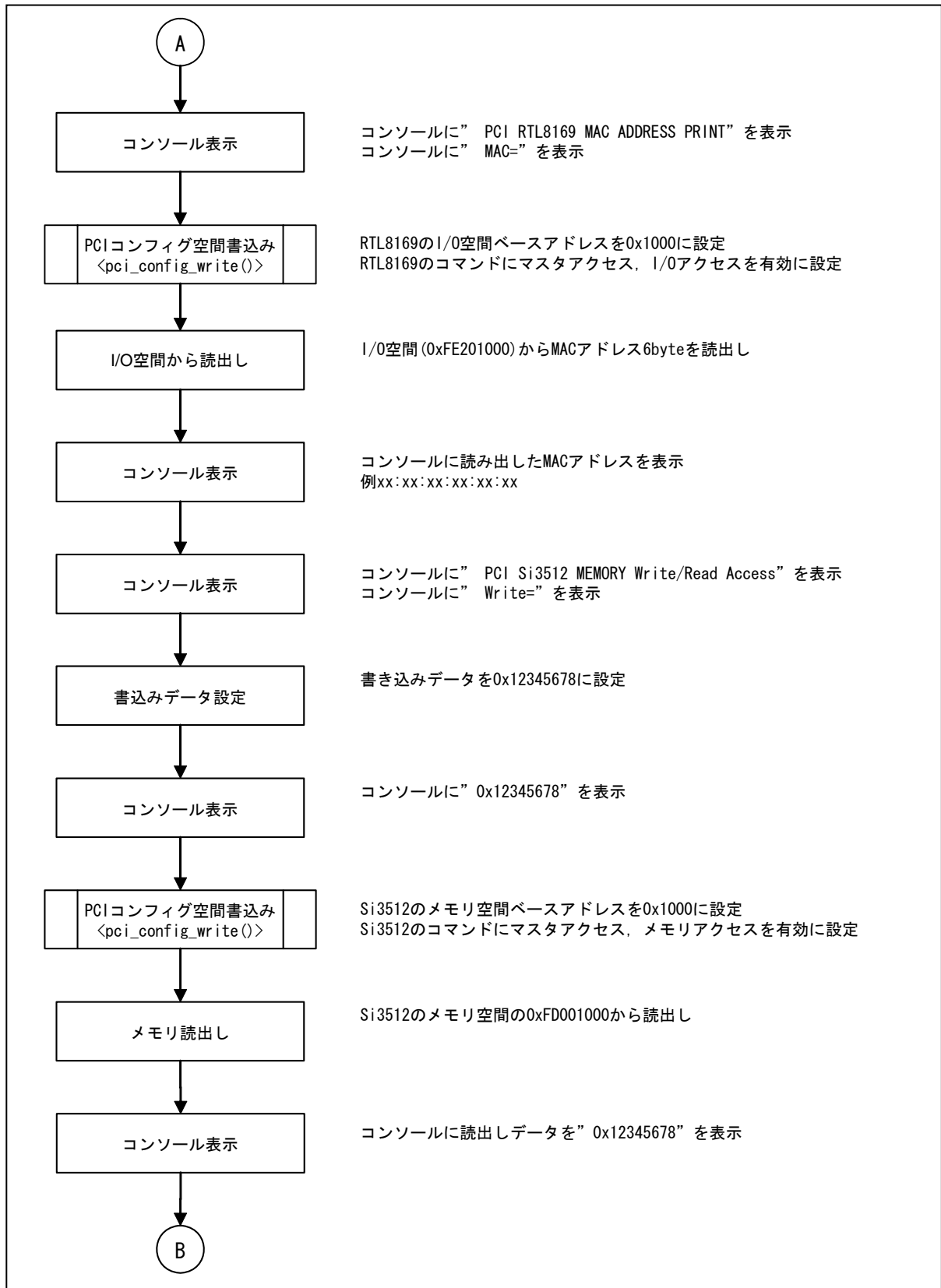


図 7 参考プログラムメインフロー2

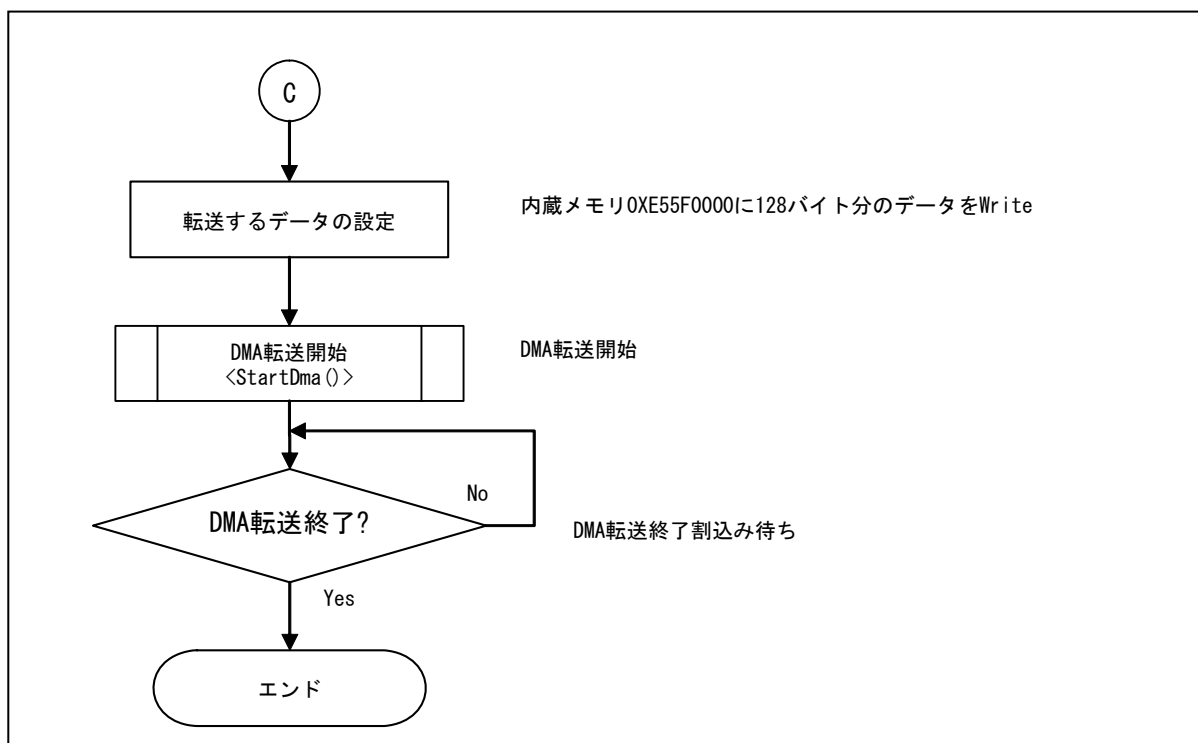


図 8 参考プログラムメインフロー3

3.4.3 ピンファンクションの初期設定

図 9, 10 にピンファンクションの初期設定のフローを示します。

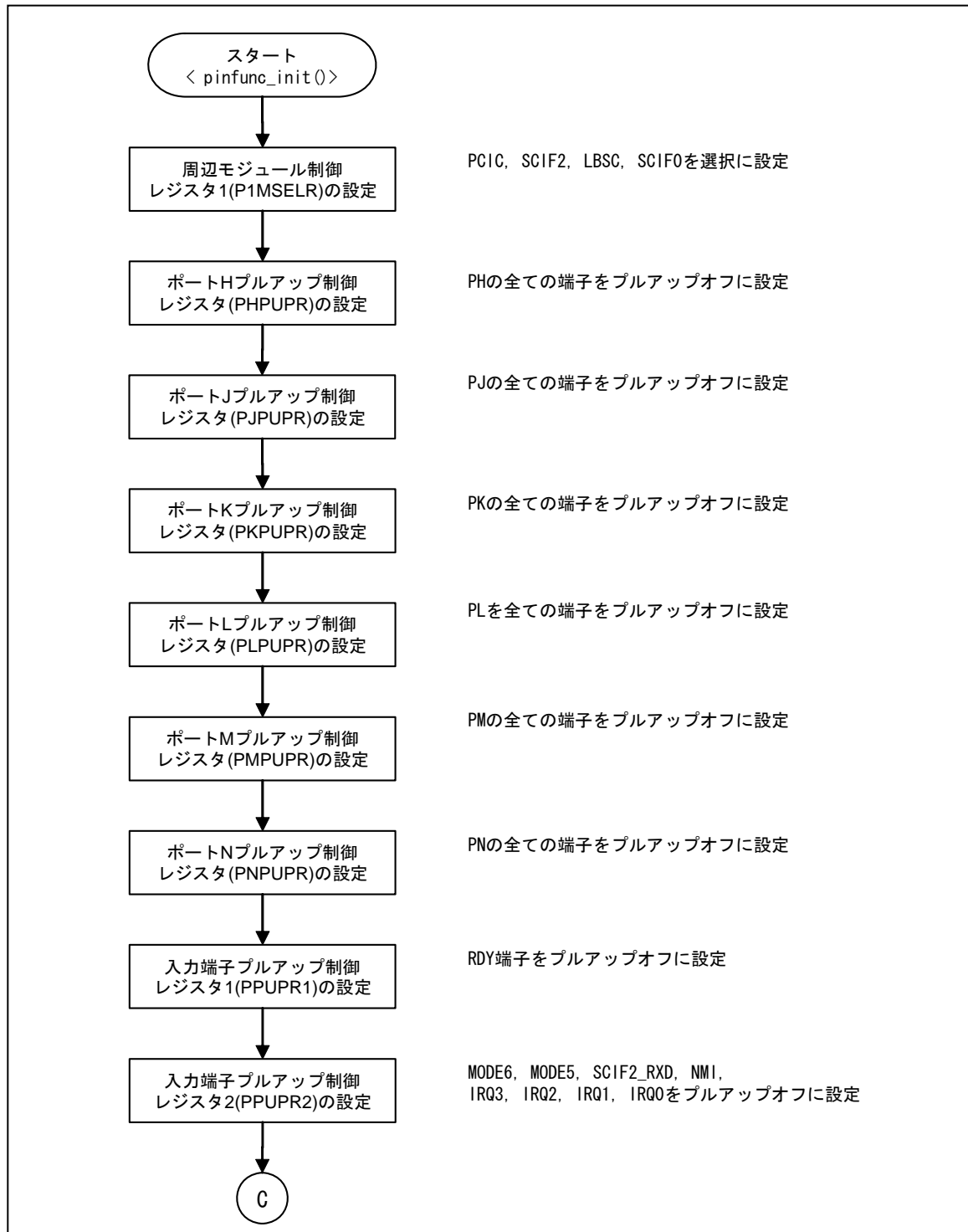


図 9 ピンファンクション初期設定フロー1

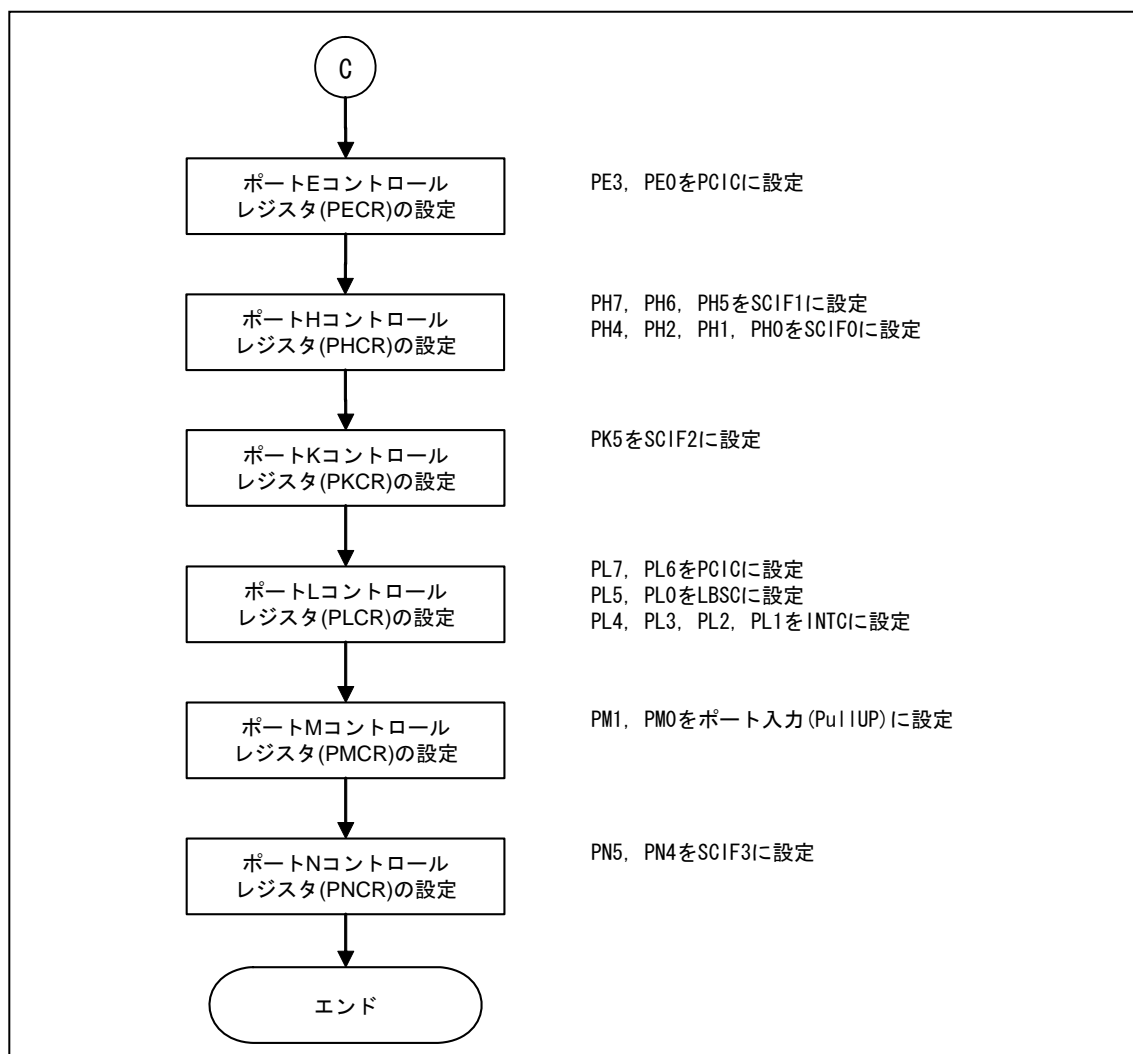


図 10 ピンファンクション初期設定フロー2

3.4.4 SCIFの初期設定

図 11 に SCIF の初期設定のフローを示します。

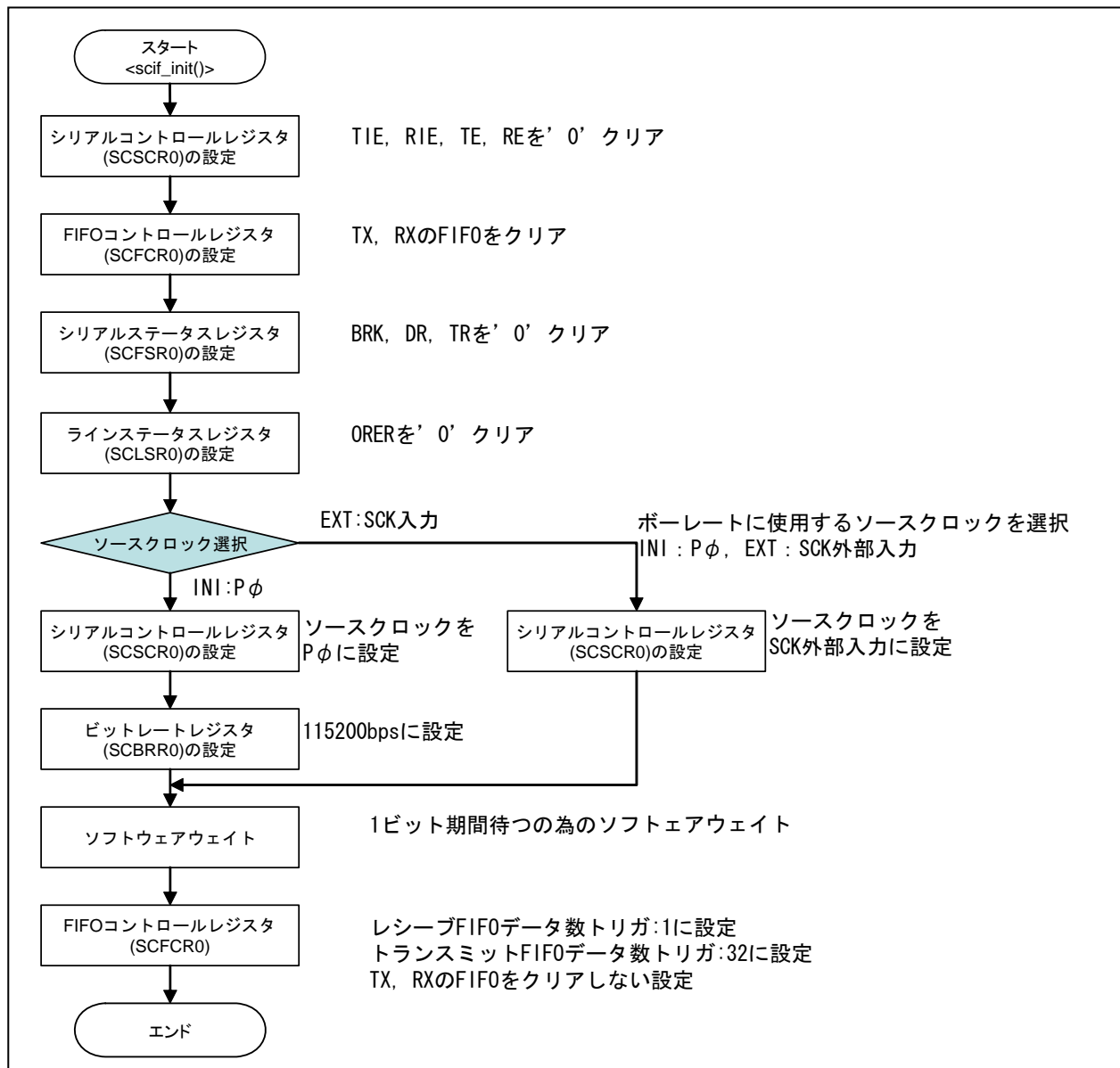


図 11 SCIF 初期設定フロー

3.4.5 PCICの初期設定

図 12 に PCIC の初期設定のフローを示します。

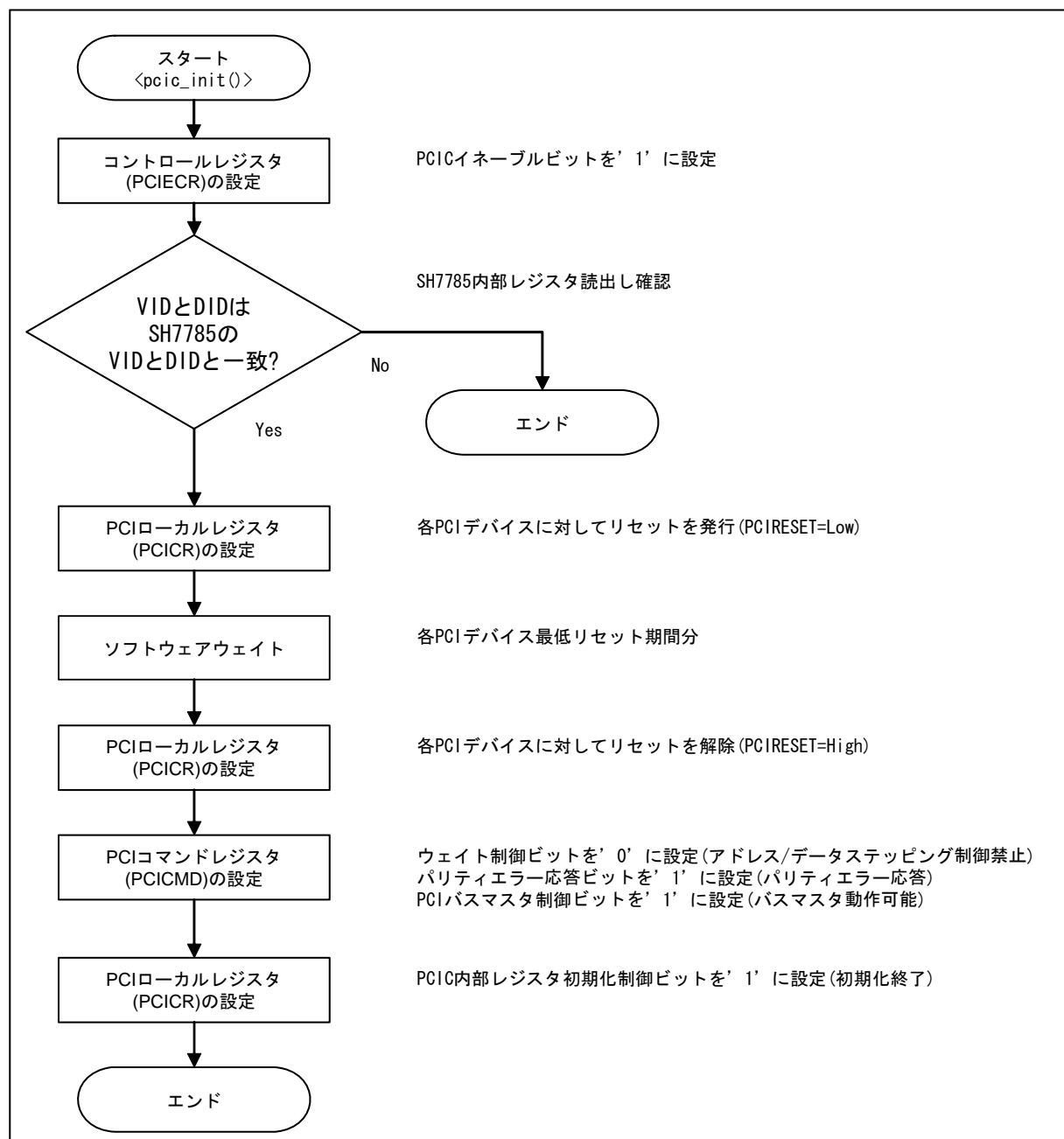


図 12 PCIC 初期設定フロー

3.4.6 コンフィグ空間読出し

図 13 にコンフィグ空間の読出しのフローを示します。

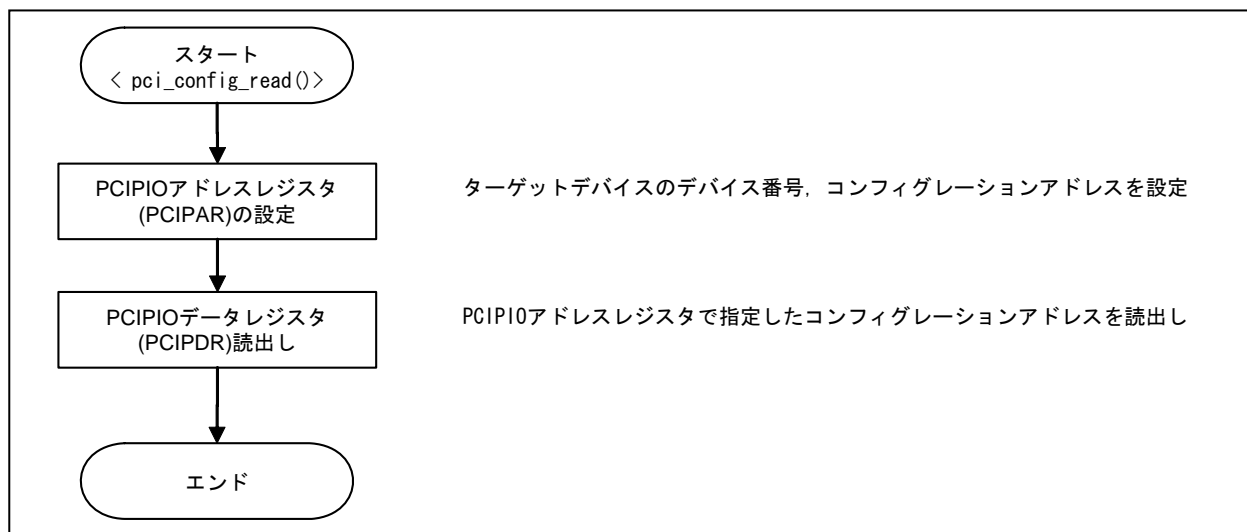


図 13 コンフィグ空間読出しフロー

3.4.7 コンフィグ空間書込み

図 14 にコンフィグ空間の書込みのフローを示します。

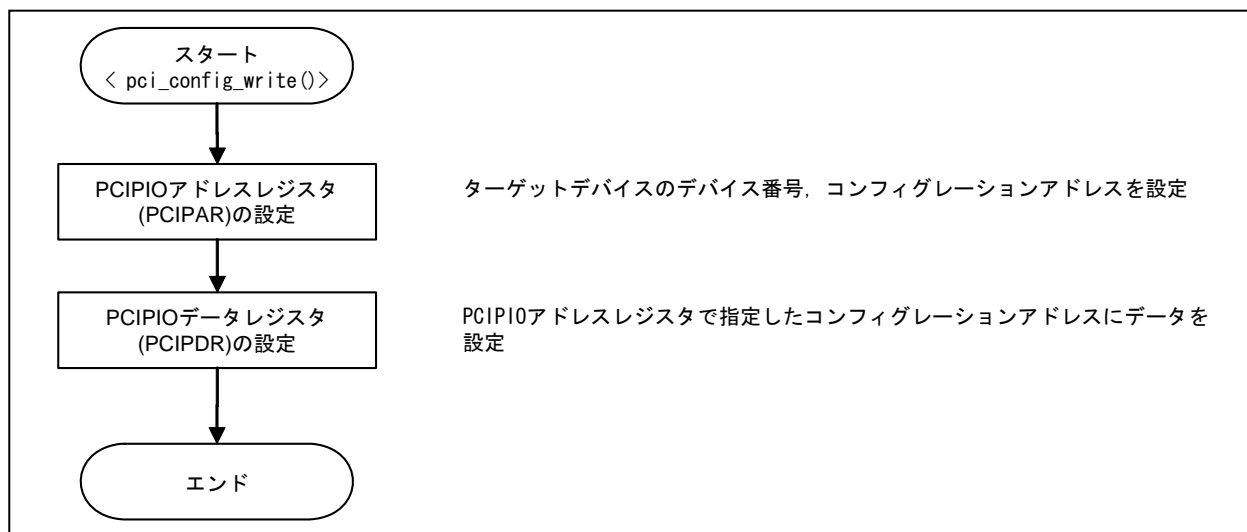


図 14 コンフィグ空間書込みフロー

3.4.8 DMA転送開始

図 15 に DMA 転送開始のフローを示します。

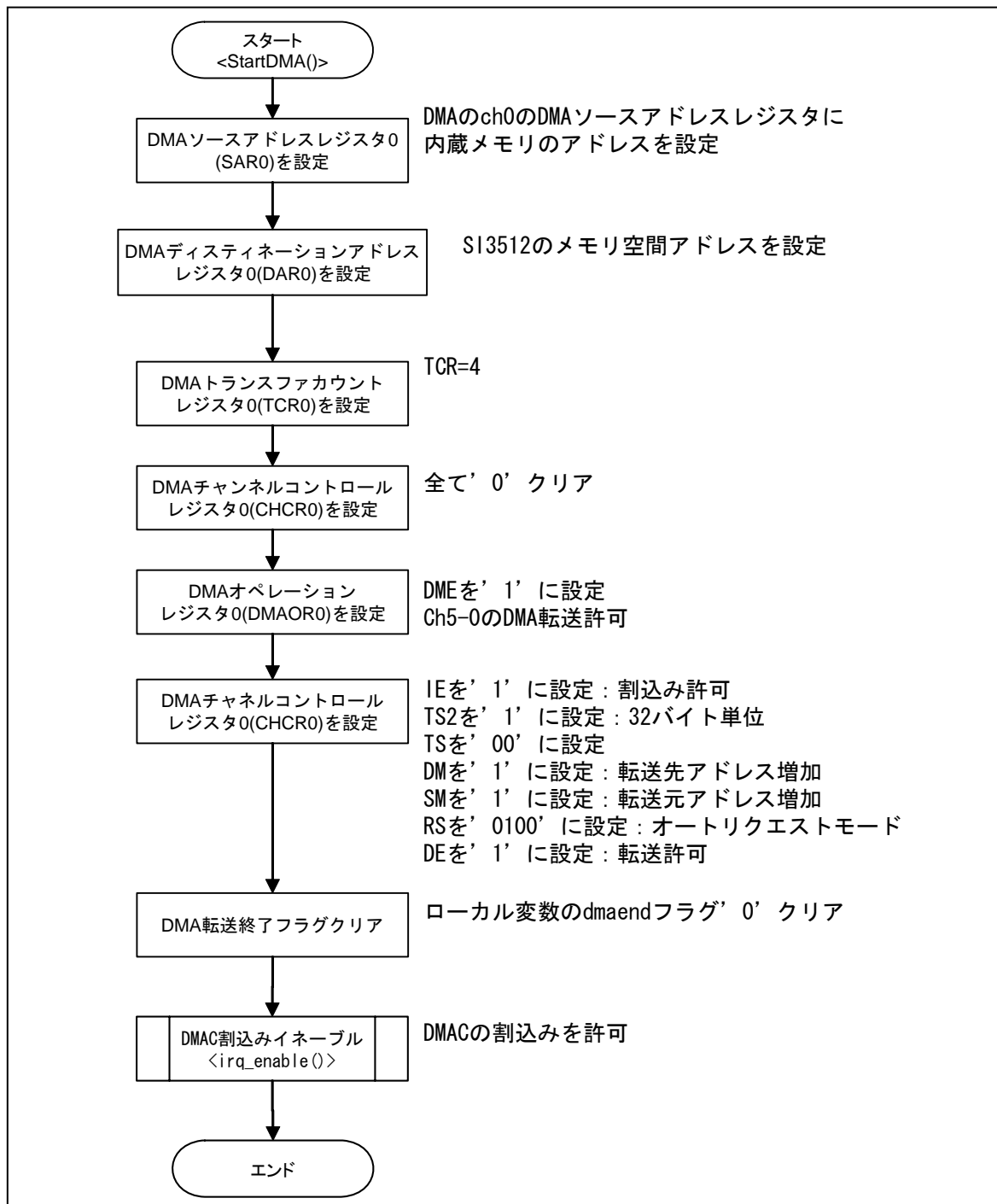


図 15 DMA 転送開始フロー

3.4.9 DMA割込み

図 16 に DMA 転送終了割込みフローを示します。

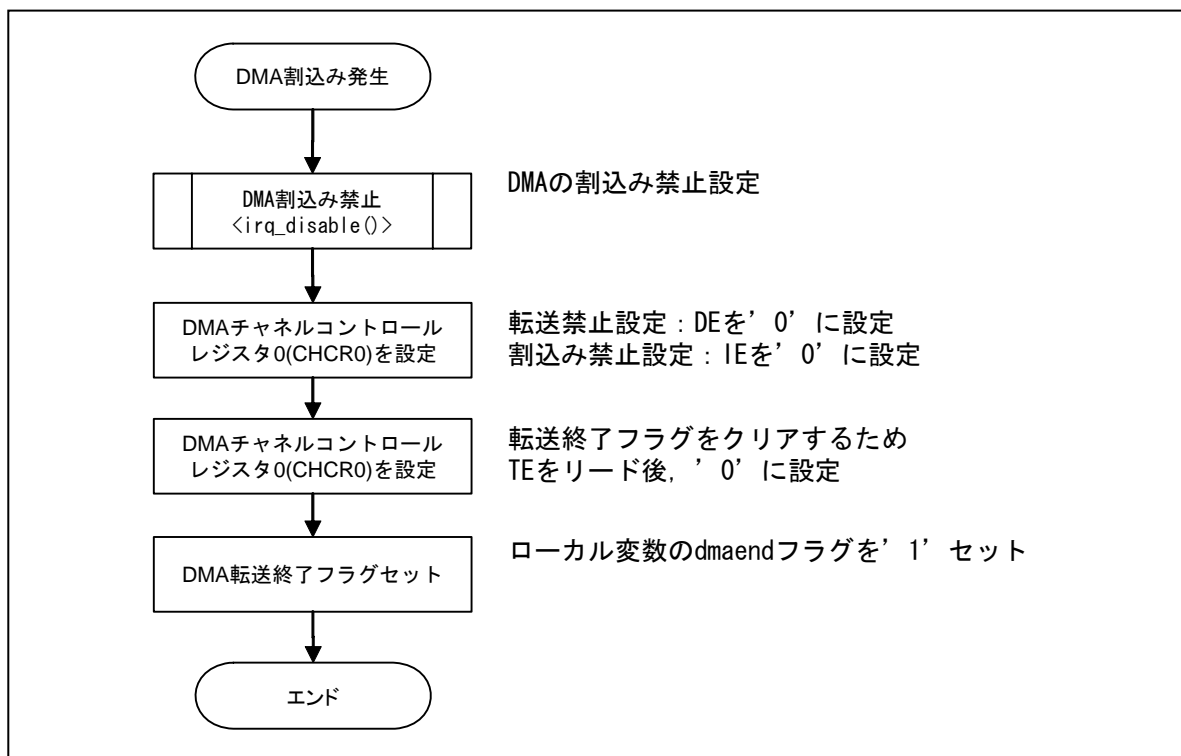


図 16 DMA 転送終了割込みフロー

3.4.10 セクション配置

表7に本応用例での各セクションの配置を示します。

表 7 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 | 0x0C000000 | |
| B | 未初期化データ領域 | RAM | | |
| R | 初期化データ領域 | RAM | | |
| S | スタック領域 | RAM | 0x0DFF8000 | |
| 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 | | |

4. 参考プログラム例

サンプルプログラムリスト”PCI_Initialize.c”

サンプルプログラムのメイン関数です。

```
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 : PCI_Initialize.c
032 * Abstract : SH7785 PCI 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 : RenesasElectronics 製 SH4A ボード 型番 R0P7785LC0011RL
039 * Description : SH7785PCI 初期設定例のサンプルプログラムです。
040 * :
041 * Operation :
042 * Limitation :
043 * :
```

```
044 ****
045 * History      : 30.SEP.2010 Ver. 1.00 First Release
046 *"FILE COMMENT END"****
047 /****
048 /*                                                    */
049 /* FILE       :PCI_Initialize.c                       */
050 /* DATE       :Fri, Oct 01, 2010                      */
051 /* DESCRIPTION :Main Program                          */
052 /* CPU TYPE   :Other                                  */
053 /*                                                    */
054 /* This file is generated by Renesas Project Generator (Ver.4.16). */
055 /*                                                    */
056 /****
057
058
059
060 #include "config.h"
061 #include "intc.h"
062 #ifdef __cplusplus
063 // #include <ios> // Remove the comment when you use ios
064 // _SINT ios_base::Init::init_cnt; // Remove the comment when you use ios
065 #endif
066
067 void main(void);
068 #ifdef __cplusplus
069 extern "C" {
070 void abort(void);
071 }
072 #endif
073
074 /* ==== 関数宣言 ==== */
075 void pinfunc_init( void );
076 static void scif_pr(int dev_no, char *title, char *data );
077 static unsigned long word_swap(unsigned long data);
078 void StartDMA( unsigned long sadd, unsigned long dadd, int cnt );
079 /* SCIF */
080 extern int scif_init(void);
081 extern void scif_transmit_data( char *Data );
082 extern void scif_transmit_data_byte( char *Data );
083 extern char scif_recive_data( char *Data );
084
085 /* PCIC */
086 extern int init_pcic(void);
087 extern unsigned long pci_read_config(unsigned int dev_no, unsigned char index);
088 extern void pci_write_config(unsigned int dev_no, unsigned char index, unsigned long data);
089
090 /* ==== 変数宣言 ==== */
```

```
091 #define   RTL8169_DEVNO   0
092 #define   SI3512_DEVNO   1
093 #define   EXTCN1_DEVNO   2
094 #define   EXTCN2_DEVNO   3
095
096 static char dev_name[16][8] = {
097   "RTL8169", "SI3512 ", "EXTCN1 ", "EXTCN2 ", "DEV4  ", "DEV5  ", "DEV6  ", "DEV7  ",
098   "DEV8  ", "DEV9  ", "DEV10 ", "DEV11 ", "DEV12 ", "DEV13 ", "DEV14 ", "DEV15 ",
099 };
100
101 static int dmaend;
102
103 /* ==== マクロ定義 ==== */
104
105
106 /*"FUNC COMMENT"*****
107 * ID          :
108 * Outline     : サンプルプログラムメイン
109 *            : (PCIC 初期設定例)
110 * Include    :
111 * Declaration : void main(void)
112 * Description : PCIC 初期設定例メイン
113 *            :
114 *            :
115 *            :
116 *            :
117 *            :
118 * Limitation :
119 *            :
120 * Argument   : none
121 * Return Value : none
122 * Calling Functions :
123 /*"FUNC COMMENT END"*****/
124 void main(void)
125 {
126   int ret,i;
127   unsigned long data;
128   unsigned long datal[128];
129   char buff[20], tmp[6];
130
131   pinfunc_init();
132   scif_init();
133   scif_transmit_data("¥rPCIC Initialize Sample Program¥n");
134
135   if(init_pcic() != 0) {
136     scif_transmit_data("¥rPCIC VENDOR_ID or DEVICE_ID Error¥n");
137     return;
```

```
138 }
139
140 scif_transmit_data("\n\rPCI DEVICE VENDER_ID & DEVICE_ID PRINT\n");
141 for(i=0;i<16;i++) {
142     data = pci_read_config(i, 0);
143     if(data==0xffffffff)
144         sprintf(buff, "%s", "Device Not Fount");
145     else
146         sprintf(buff, "%x", word_swap(data));
147     scif_pr(i, dev_name[i], buff);
148 }
149
150 scif_transmit_data("\n\rPCI RTL8169 MAC ADDRESS PRINT\n");
151 scif_transmit_data("\rMAC = ");
152 pci_write_config(RTL8169_DEVNO, 0x10, 0x00001000);
153 pci_write_config(RTL8169_DEVNO, 0x04, 0x00000005);
154
155 data = (*(unsigned long *)0xFE201000);
156 for(i=0;i<4;i++)
157     tmp[i] = (unsigned char)(data >> 8*i);
158
159 data = (*(unsigned long *)0xFE201004);
160 for(i=4;i<6;i++)
161     tmp[i] = (unsigned char)(data >> 8*(i-4));
162
163 for(i=0;i<5;i++) {
164     sprintf(buff, "%02x", (tmp[i] & 0x000000FF));
165     scif_transmit_data(buff);
166     scif_transmit_data(":");
167 }
168 sprintf(buff, "%02x", (tmp[5] & 0x000000FF));
169 scif_transmit_data(buff);
170 scif_transmit_data("\n");
171
172 scif_transmit_data("\n\rPCI Si3512 MEMORY Write/Read ACCESS\n");
173 scif_transmit_data("\rWrite Data = ");
174 data = 0x12345678;
175 sprintf(buff, "%02x", data);
176 scif_transmit_data(buff);
177 scif_transmit_data("\n");
178
179 pci_write_config(SI3512_DEVNO, 0x24, 0x00001000);
180 pci_write_config(SI3512_DEVNO, 0x04, 0x00000006);
181 (*(unsigned long *)0xFD00104C) = data;
182 data = 0;
183 data = (*(unsigned long *)0xFD00104C);
184
```

```
185  scif_transmit_data("¥rRead Data = ");
186  sprintf(buff, "%02x", data);
187  scif_transmit_data(buff);
188  scif_transmit_data("¥n");
189
190  for(i=0;i<128;i++)
191      (*(unsigned char *) (0xe55fe000 + i)) = i;
192  /* Burst Read 32Byte * 2 */
193  StartDMA( 0xe55fe000, 0xfd001000, 4 );
194  while(!dmaend);
195
196
197 }
198
199 /*"FUNC COMMENT"*****
200 * ID          :
201 * Outline     : サンプルプログラムメイン
202 *             : (PCIC 初期設定例)
203 * Include    :
204 * Declaration : void pinfunc_init( void )
205 * Description : ピンファンクションの設定
206 *             :
207 *             :
208 *             :
209 *             :
210 *             :
211 * Limitation  :
212 *             :
213 * Argument    : none
214 * Return Value : none
215 * Calling Functions :
216 *"FUNC COMMENT END"*****/
217
218 void pinfunc_init( void )
219 {
220     GPIO.P1MSELR.WORD = 0x3780;
221     GPIO.PHPUPR.BYTE = 0x00;
222     GPIO.PJPUPR.BYTE = 0x00;
223     GPIO.PKPUPR.BYTE = 0x00;
224     GPIO.PLPUPR.BYTE = 0x00;
225     GPIO.PMPUPR.BYTE = 0xFC;
226     GPIO.PNPUPR.BYTE = 0x00;
227     GPIO.PPUPR1.WORD = 0xFFFFB;
228     GPIO.PPUPR2.WORD = 0xFF00;
229     GPIO.PECR.WORD = 0x0000;
230     GPIO.PHCR.WORD = 0x00C0;
231     GPIO.PKCR.WORD = 0x03FF;
```

```

232  GPIO.PLCR.WORD = 0x0000;
233  GPIO.PMCR.WORD = 0xFFFF;
234  GPIO.PNCR.WORD = 0xF0FF;
235 }
236
237 /*"FUNC COMMENT"*****
238 * ID          :
239 * Outline     : サンプルプログラムメイン
240 *             : (PCIC 初期設定例)
241 * Include     :
242 * Declaration : static void scif_pr(int dev_no, char *title, char *data )
243 * Description : シリアルコンソールデバイス&データ表示
244 *             :
245 *             :
246 *             :
247 *             :
248 *             :
249 * Limitation  :
250 *             :
251 * Argument    : dev_no : デバイス番号, *title : デバイス名,
252 *             : *data  : 表示データ
253 * Return Value : none
254 * Calling Functions :
255 /*"FUNC COMMENT END"*****/
256 static void scif_pr(int dev_no, char *title, char *data )
257 {
258   scif_transmit_data("¥r");
259   scif_transmit_data(title);
260   scif_transmit_data(" = ");
261   scif_transmit_data(data);
262   scif_transmit_data("¥n");
263 }
264
265 /*"FUNC COMMENT"*****
266 * ID          :
267 * Outline     : サンプルプログラムメイン
268 *             : (PCIC 初期設定例)
269 * Include     :
270 * Declaration : static unsigned long word_swap(unsigned long data)
271 * Description : ロングデータワードスワップ
272 *             :
273 *             :
274 *             :
275 *             :
276 *             :
277 * Limitation  :
278 *             :

```

```

279 * Argument      : data : データ
280 * Return Value  : 変換後データ
281 * Calling Functions :
282 *"FUNC COMMENT END"*****
283 static unsigned long word_swap(unsigned long data)
284 {
285     unsigned long tmp;
286     tmp = ((data & 0xFFFF0000) >> 16) | ((data & 0x0000FFFF) << 16);
287     return tmp;
288 }
289
290 /*"FUNC COMMENT"*****
291 * ID              :
292 * Outline         : サンプルプログラムメイン
293 *                : (PCIC 初期設定例)
294 * Include        :
295 * Declaration    : void StartDMA( unsigned long sadd, unsigned long dadd, int cnt )
296 * Description    : DMAC-ch0 の初期設定及び DMA 転送スタート
297 *                : DMA の割込み許可
298 *                :
299 *                :
300 *                :
301 *                :
302 * Limitation     :
303 *                :
304 * Argument       : sadd:転送元アドレス, fdadd:転送先アドレス, cnt:転送回数
305 * Return Value   : none
306 * Calling Functions :
307 *"FUNC COMMENT END"*****
308 void StartDMA( unsigned long sadd, unsigned long dadd, int cnt )
309 {
310     *(unsigned long *)0xFC808020 = sadd;
311     *(unsigned long *)0xFC808024 = dadd;
312
313     DMACO.TCR.BIT.CNT      = cnt;
314     DMACO.CHCR.LONG       = 0;
315     DMAC.DMAOR0.BIT.DME   = 1;    /* DMAC0-5 許可 */
316     DMACO.CHCR.BIT.IE     = 1;    /* 割込み許可 */
317     DMACO.CHCR.BIT.TS2    = 1;    /* 32Byte アクセス */
318     DMACO.CHCR.BIT.TS     = 0;    /* 32Byte アクセス */
319     DMACO.CHCR.BIT.DM     = 1;    /* 転送先増加 */
320     DMACO.CHCR.BIT.SM     = 1;    /* 転送元増加 */
321     DMACO.CHCR.BIT.RS     = 4;    /* オートリクエスト */
322     DMACO.CHCR.BIT.DE     = 1;    /* 転送許可 */
323     dmaend = 0;
324     irq_enable( _DMACO );
325 }

```

```
326
327 void dmac0_irq( void )
328 {
329     int tmp;
330     DMACO.CHCR.BIT.IE      = 0;    /* 割込み禁止 */
331     DMACO.CHCR.BIT.DE      = 0;    /* 転送禁止 */
332     tmp = DMACO.CHCR.BIT.TE;
333     DMACO.CHCR.BIT.TE      = 0;    /* フラグクリア */
334     dmaend = 1;
335 }
336
337 #ifdef __cplusplus
338 void abort(void)
339 {
340
341 }
342 #endif
```


サンプルプログラムリスト”pcic.c”

PCIC の初期設定例です。

```
001 /*****
002 * DISCLAIMER
003
004 * This software is supplied by Renesas Electronics Corp. and is only
005 * intended for use with Renesas products. No other uses are authorized.
006
007 * This software is owned by Renesas Electronics Corp. 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 * TECHNOLOGY CORP. 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 Corp., All Rights Reserved. */
029 /*"FILE COMMENT"***** Technical reference data *****
030 * System Name : SH7785 Sample Program
031 * File Name : pcic.c
032 * Abstract : SH7785 PCIC 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 : RenesasElectronics 製 SH-4A ボード 型番 R0P7785LC0011RL
039 * Description : SH7785 PCIC 設定例のサンプルプログラムです。
040 * :
041 * Operation :
042 * Limitation :
043 * :
044 *****/
045 * History : 30.SEP.2010 Ver. 1.00 First Release
```

```

046 *"FILE COMMENT END"*****/
047
048 #include "config.h"
049 #include "pcic.h"
050
051 extern void delay( int cnt );
052
053 /*"FUNC COMMENT"*****
054 * ID          :
055 * Outline     : サンプルプログラムメイン
056 *            : (PCIC 初期設定例)
057 * Include     :
058 * Declaration : int init_pcic(void)
059 * Description : PCIC の初期設定
060 *            : VENDER_ID 及び DEVICE_ID の判定
061 *            : VENDER_ID 及び DEVICE_ID の判定
062 *            : PCI バスマスタ設定
063 *            :
064 *            :
065 * Limitation  :
066 *            :
067 * Argument    : none
068 * Return Value : -1 : VENDER_ID 又は DEVICE_ID の不一致
069 * Calling Functions :
070 *"FUNC COMMENT END"*****/
071 int init_pcic(void)
072 {
073     PCIC.PCIECR.BIT.ENBL = 1;
074     if( (PCIC.PCIVID != VENDER_ID)
075         && PCIC.PCIDID != DEVICE_ID)
076         return -1;
077
078     /* Toggle PCI reset pin */
079     PCIC.PCICR.LONG = PCICR_PREFIX | PCICR_PRST;
080     delay(10000);
081     PCIC.PCICR.LONG = PCICR_PREFIX;
082     PCIC.PCICMD.BIT.WCC = 0;      /* アドレス/データステッピング制御禁止 */
083     PCIC.PCICMD.BIT.PER = 1;     /* パリティエラー応答 */
084     PCIC.PCICMD.BIT.BM = 1;     /* PCI バスマスタ */
085
086     PCIC.PCICR.LONG = PCICR_PREFIX | PCICR_CFIN;
087
088     return 0;
089 }
090
091 /*"FUNC COMMENT"*****
092 * ID          :

```

```

093 * Outline      : サンプルプログラムメイン
094 *              : (PCIC 初期設定例)
095 * Include      :
096 * Declaration  : unsigned long pci_read_config(unsigned int dev_no, unsigned char index)
097 * Description  : ターゲットデバイスの PCI コンフィグレーションデータを読み出し
098 *              :
099 *              :
100 *              :
101 *              :
102 *              :
103 * Limitation   :
104 *              :
105 * Argument     : dev_no : デバイス番号, index : コンフィグレーションアドレス
106 * Return Value : 読み出しデータ
107 * Calling Functions :
108 * "FUNC COMMENT END"*****/
109 unsigned long pci_read_config(unsigned int dev_no, unsigned char index)
110 {
111     unsigned long par_data = 0x80000000 | (dev_no << 11);
112
113     PCIC.PCIPAR.LONG = par_data | (index & 0xfc);
114     return PCIC.PCIPDR;
115 }
116
117 /*"FUNC COMMENT"*****
118 * ID           :
119 * Outline      : サンプルプログラムメイン
120 *              : (PCIC 初期設定例)
121 * Include      :
122 * Declaration  : void pci_write_config(unsigned int dev_no, unsigned char index, unsigned long data)
123 * Description  : ターゲットデバイスの PCI コンフィグレーションにデータを書き込む
124 *              :
125 *              :
126 *              :
127 *              :
128 *              :
129 * Limitation   :
130 *              :
131 * Argument     : dev_no : デバイス番号, index : コンフィグレーションアドレス, data : 書き込みデータ
132 * Return Value : none
133 * Calling Functions :
134 * "FUNC COMMENT END"*****/
135 void pci_write_config(unsigned int dev_no, unsigned char index, unsigned long data)
136 {
137     unsigned long par_data = 0x80000000 | (dev_no << 11);
138     PCIC.PCIPAR.LONG = par_data | (index & 0xfc);
139     PCIC.PCIPDR = data;

```

```
140 }
```

サンプルプログラムリスト”pcic.h”

“pcic.c”で使用するヘッダーです。

```
01 #ifndef _PCIC_H
02 #define _PCIC_H
03
04
05 #define VENDER_ID 0x1912
06 #define DEVICE_ID 0x0007
07
08 /* SH7785 PCI Local Registers */
09 #define PCICR_PREFIX          0xA5000000    /* CR prefix for write */
10 #define PCICR_PRST           0x00000002    /* PCI Reset Assert */
11 #define PCICR_CFIN           0x00000001    /* Central Fun. Init Done */
12 #define PCICR_PFCS           0x00000800    /* TRDY/IRDY Enable */
13 #define PCICR_FTO            0x00000400    /* TRDY/IRDY Enable */
14 #define PCICR_PFE            0x00000200    /* Target Read Single */
15 #define PCICR_TBS            0x00000100    /* Target Byte Swap */
16 #define PCICR_ARBM           0x00000040    /* PCI Arbitration Mode */
17 #define PCICR_IOCS           0x00000004    /* INTA output assert */
18
19 #endif /* _PCIC_H */
```

サンプルプログラムリスト”scif.c”

シリアルコミュニケーションチャンネル 0 の初期設定例です。

```
001 /*****
002 * DISCLAIMER
003
004 * This software is supplied by Renesas Electronics Corp. and is only
005 * intended for use with Renesas products. No other uses are authorized.
006
007 * This software is owned by Renesas Electronics Corp. 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 * TECHNOLOGY CORP. 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 Corp., All Rights Reserved. */
029 /*"FILE COMMENT"***** Technical reference data *****
030 * System Name : SH7785 Sample Program
031 * File Name : scif.c
032 * Abstract : SH7785 SCIF 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 : RenesasElectronics 製 SH-4A ボード 型番 R0P7785LC0011RL
039 * Description : SH7785SCIF 設定例のサンプルプログラムです。
040 * :
041 * Operation :
042 * Limitation :
043 * :
044 *****/
045 * History : 30.SEP.2010 Ver. 1.00 First Release
```

```
046 *"FILE COMMENT END"*****/  
047  
048  
049 #include "scif.h"  
050  
051 /*"FUNC COMMENT"*****  
052 * ID :  
053 * Outline : サンプルプログラムメイン  
054 * : (PCIC 初期設定例)  
055 * Include :  
056 * Declaration : int delay( int cnt )  
057 * Description : ソフトウェアウェイト  
058 * : cnt 分 for 文を回す  
059 * :  
060 * :  
061 * :  
062 * :  
063 * Limitation :  
064 * :  
065 * Argument : cnt  
066 * Return Value : none  
067 * Calling Functions :  
068 *"FUNC COMMENT END"*****/  
069 void delay( int cnt )  
070 {  
071     int i;  
072     for(i=0;i<cnt;i++);  
073 }  
074  
075 /*"FUNC COMMENT"*****  
076 * ID :  
077 * Outline : サンプルプログラムメイン  
078 * : (PCIC 初期設定例)  
079 * Include :  
080 * Declaration : int scif_init(void)  
081 * Description : SCIF の初期設定  
082 * :  
083 * :  
084 * :  
085 * :  
086 * :  
087 * Limitation :  
088 * :  
089 * Argument : none  
090 * Return Value : -1 : ボーレートクロック計算エラー  
091 * Calling Functions :  
092 *"FUNC COMMENT END"*****/
```

```

093 int scif_init(void)
094 {
095     unsigned short data;
096     int t = -1, cnt = 0;
097
098     SCIF.SCSCR.WORD = 0x0000;      /* TIE, RIE, TE, RE Clear */
099
100     SCIF.SCFR.BIT.TFCL = 1;      /* Tx FIFO Clear */
101     SCIF.SCFR.BIT.RFCL = 1;      /* Rx FIFO Clear */
102
103     SCIF.SCFSR.WORD = 0x0000;    /* BRK, DR, TR Clear */
104
105 #if defined(CONFIG_SCIF_CLK_EXTERNAL)
106     SCIF.SCSCR.BIT.CKE = 2; /* クロックソース : SCK */
107 #elif defined(CONFIG_SCIF_CLK_PCLK)
108     SCIF.SCSCR.BIT.CKE = 0; /* クロックソース : PCLK */
109     t = SCBRR_VALUE(CONFIG_BPS, CONFIG_SCIF_CLK_PCLK);
110 #endif /* CONFIG_SCIF_CLK */
111
112     if(t > 0) {
113         while(t >= 256) {
114             cnt++;
115             t >> 2;
116         }
117         if(cnt > 3)
118             return -1;
119
120         SCIF.SCSMR.BIT.CKS = cnt;
121         SCIF.SCBRR = t;
122     }
123     delay(1000);
124
125     SCIF.SCFR.BIT.RTRG = 0;
126     SCIF.SCFR.BIT.TTRG = 0;
127     SCIF.SCFR.BIT.TFCL = 1;      /* Tx FIFO Clear */
128     SCIF.SCFR.BIT.RFCL = 1;      /* Rx FIFO Clear */
129
130     SCIF.SCFR.BIT.TFCL = 0;      /* Tx FIFO Not Clear */
131     SCIF.SCFR.BIT.RFCL = 0;      /* Rx FIFO Not Clear */
132     SCIF.SCSCR.BIT.TE = 1;
133     SCIF.SCSCR.BIT.RE = 1;
134     return 0;
135 }
136
137 /*"FUNC COMMENT"*****
138 * ID :
139 * Outline : サンプルプログラムメイン

```

```

140 *          : (PCIC 初期設定例)
141 * Include      :
142 * Declaration  : void scif_transmit_data( char *Data )
143 * Description  : SCIF の複数 Byte データ送信
144 *          :
145 *          :
146 *          :
147 *          :
148 *          :
149 * Limitation   :
150 *          :
151 * Argument     : *Data : 送信データ格納
152 * Return Value : none
153 * Calling Functions :
154 *""FUNC COMMENT END""*****/
155 void      scif_transmit_data( char      *Data )
156 {
157     while( *Data )
158     {
159         while(!(SCIF.SCFSR.BIT.TDFE)); /* 送信データ書き込み許可状態になるまでウェイト */
160         SCIF.SCFSTR = *Data;          /* 送信データ設定          */
161         Data++;
162         while(!(SCIF.SCFSR.BIT.TEND)); /* 送信終了待ち */
163         SCIF.SCFSR.BIT.TDFE = 0;
164         SCIF.SCFSR.BIT.TEND = 0;
165     }
166 }
167
168 /*""FUNC COMMENT""*****
169 * ID          :
170 * Outline     : サンプルプログラムメイン
171 *          : (PCIC 初期設定例)
172 * Include     :
173 * Declaration : void scif_transmit_byte_data( char *Data )
174 * Description : SCIF の 1Byte データ送信
175 *          :
176 *          :
177 *          :
178 *          :
179 *          :
180 * Limitation  :
181 *          :
182 * Argument    : *Data : 送信データ格納
183 * Return Value : none
184 * Calling Functions :
185 *""FUNC COMMENT END""*****/
186 void      scif_transmit_data_byte( char      *Data )

```



```

187 {
188     while(! (SCIF.SCFSR.BIT.TDFE)); /* 送信データ書き込み許可状態になるまでウェイト */
189     SCIF.SCFSTR = *Data; /* 送信データ設定 */
190     while(! (SCIF.SCFSR.BIT.TEND)); /* 送信終了待ち */
191     SCIF.SCFSR.BIT.TDFE = 0;
192     SCIF.SCFSR.BIT.TEND = 0;
193 }
194
195 /*"FUNC COMMENT"*****
196 * ID :
197 * Outline : サンプルプログラムメイン
198 * : (PCIC 初期設定例)
199 * Include :
200 * Declaration : char scif_recive_data( char *Data )
201 * Description : SCIF のデータ受信
202 * :
203 * :
204 * :
205 * :
206 * :
207 * Limitation :
208 * :
209 * Argument : *Data : 受信データ格納
210 * Return Value : -1 : 受信エラー
211 * Calling Functions :
212 /*"FUNC COMMENT END"*****/
213 char scif_recive_data( char *Data )
214 {
215     unsigned char ReadData, i = 0;
216     char ret_cd = 0;
217
218     for(;;)
219     {
220         if(( SCIF.SCFSR.BIT.ER ) ||
221            ( SCIF.SCFSR.BIT.BRK ) ||
222            ( SCIF.SCFSR.BIT.DR )) /* エラー発生? */
223         {
224             ReadData = SCIF.SCFRDR; /* データダミー読み込み */
225             ret_cd = -1; /* 受信エラー設定 */
226             SCIF.SCFSR.WORD &= 0x0000; /* エラークリア */
227             SCIF.SCLSR.WORD &= 0x0000;
228         }
229         else if( SCIF.SCFSR.BIT.RDF ) /* データ受信? */
230         {
231             *Data = SCIF.SCFRDR; /* データ取得 */
232             SCIF.SCFSR.BIT.RDF = 0; /* 受信サインクリア */
233             SCIF.SCFSR.BIT.DR = 0; /* 受信サインクリア */

```

```
234         scif_transmit_data_byte( Data );
235         if( *Data == '\n' )      /* 取得データが CR?    */
236         {
237             break; /* 処理終了    */
238         }
239         if( *Data == 0x0d )      /* 取得データが CR?    */
240         {
241             break; /* 処理終了    */
242         }
243         Data++; /* データ取得アドレス次設定    */
244         if( ++i == 4 )
245         {
246             ret_cd = -1;
247         }
248     }
249     if( ret_cd == -1 )
250     {
251         break;
252     }
253 }
254 return( ret_cd );
255 }
256
257
```

サンプルプログラムリスト”scif.h”

“scif.c”で使用するヘッダーです。

```
01
02 #ifndef _SCIF_H
03 #define _SCIF_H
04
05 #include "config.h"
06
07 #if defined(CONFIG_SCIF0)
08 #define SCIF (*(volatile struct st_scif *)0xFFEA0000) /* SCIF0 Address */
09 #elif defined(CONFIG_SCIF1)
10 #define SCIF (*(volatile struct st_scif *)0xFFEB0000) /* SCIF1 Address */
11 #elif defined(CONFIG_SCIF2)
12 #define SCIF (*(volatile struct st_scif *)0xFFEC0000) /* SCIF2 Address */
13 #elif defined(CONFIG_SCIF3)
14 #define SCIF (*(volatile struct st_scif *)0xFFED0000) /* SCIF3 Address */
15 #elif defined(CONFIG_SCIF4)
16 #define SCIF (*(volatile struct st_scif *)0xFFEE0000) /* SCIF4 Address */
17 #elif defined(CONFIG_SCIF5)
18 #define SCIF (*(volatile struct st_scif *)0xFFEF0000) /* SCIF5 Address */
19 #endif /* CONFIG_SCIFn */
20
21 // #define SCBRR_VALUE(bps, clk) ((clk+16*bps)/(16*bps)-1)
22 #define SCBRR_VALUE(bps, clk) ((clk)/(32*bps)-1)
23
24 /* SCFCR */
25 #define RTRG1 0
26 #define RTRG16 1
27 #define RTRG32 2
28 #define RTRG48 3
29 #define TTRG32 0
30 #define TTRG16 1
31 #define TTRG2 2
32 #define TTRG0 3
33
34
35
36 #endif /* _SCIF_H */
```

5. 実行結果

上記プログラムを実行すると、

- PCI の各デバイスの VendorID と DeviceID がコンソールに表示
- RTL8169 の MAC アドレスがコンソールに表示

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

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

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



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

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

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

(03)5201-5307

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