

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

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

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

2010年4月1日

ルネサスエレクトロニクス株式会社

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

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

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無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。

## SH7780 グループ

### DMAC メモリ間転送例 (LRAM      DDR SDRAM)

---

#### 要旨

この資料は、SH7780 グループのメモリ間転送 (LRAM → DDR SDRAM) の設定方法例、および応用例について説明しています。

#### 動作確認デバイス

SH7780 (日立超 LSI システムズ製 Solution Engine MS7780SE03)

#### 目次

1. はじめに .....	2
2. 応用例の説明 .....	3
3. 参考プログラムリスト .....	11
4. 参考ドキュメント .....	22

## 1. はじめに

### 1.1 仕様

DMA チャンネル 0 を使用して、L メモリページ 0、ページ 1 (16 KB) に格納されているデータ全てを DDR-SDRAM に転送します。5 ms ごとに転送終了を確認し転送が終了している場合は、再度 DMA 転送を再起動します。

バスモードはサイクルスチールモードを、DMA 転送要求にはオートリクエストモード (ソフトウェアによる転送要求) を使用します。

### 1.2 使用機能

- DMAC チャンネル 0

### 1.3 適用条件

- マイコン: SH7780
- 動作周波数:
  - 内部クロック: 400 MHz
  - SuperHyway クロック: 200 MHz
  - 周辺クロック: 33 MHz
  - DDR クロック: 160 MHz
  - 外部バスクロック: 33 MHz
  - PCI バスクロック: 33 MHz
- クロック動作モード: モード 3 (MODE7 = Low, MODE2 = Low, MODE1 = High, MODE0 = High)
- データアライメント: リトルエンディアン
- アドレスモード: 29 ビットアドレスモード
- C コンパイラ: ルネサス テクノロジ製  
SuperH RISC engine ファミリ C/C++コンパイラパッケージ Ver.9.1.0

### 1.4 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、アプリケーションノート「SH7780 初期設定例」(RJJ06B0860-0100)、「TMU 動作例」(一定同期タイマによる 1 秒の生成)(RJJ06B0871-0100) を使用した設定条件で動作確認をしています。そちらも合わせて参照してください。

## 2. 応用例の説明

本プログラム例では、ダイレクトメモリアクセスコントローラ (DMAC) チャンネル 0 を使用し、内蔵 RAM (LRAM) から外部メモリ (DDR-SDRAM) へのブロック転送を行ないます。初回の転送開始後は 5 ms ごとに転送が終了しているか確認を行ない、転送終了している場合は転送を再度開始します。一ブロック分の転送が終わった後は転送を禁止して 5 ms 経過するまで待機します。また転送中にアドレスエラーが発生した場合には、転送終了後と同様に待機します。

### 2.1 参考プログラムの説明

本プログラム例は

- (1) dmac.c
- (2) main.c
- (3) intprg.c
- (4) vecttbl.src

の 4 つのソースプログラムから構成されます。また TMU 動作例の 5 ms カウントも使用しますので、TMU 動作例の tmu.c もプロジェクトに追加する必要があります。

- (1) dmac.c は、本プログラムで使用する DMAC の動作設定関数とアドレスエラーが発生した場合に、再度 DMAC 転送を開始させるリスタート関数を記述しています。  
本プログラム例のベースとなる初期設定例には含まれていません。
- (2) main.c では、ステータスレジスタ (SR) の設定と DMAC の動作設定関数コールを行なっています。  
初期設定例に含まれる main.c を本プログラム例に合わせて変更してください。
- (3) intprg.c では例外/割り込みハンドラよりコールされる割り込みプログラムを記述しています。  
初期設定例に含まれる intprg.c を本プログラム例に合わせて変更してください。また、TMU 動作例の 5 ms カウントも合わせて使用しますので、TMU の割り込みプログラムも合わせて変更する必要があります。
- (4) vecttbl.src ではリセットベクタテーブル、例外/割り込みベクタテーブル、割り込みマスクテーブルを記述しています。受け付けた割り込みを繰り返し受け付けないように、ステータスレジスタの IMASK ビットに設定する割り込みマスクレベルを割り込みマスクテーブルに記述する必要があります。

初期設定例に含まれる vecttbl.src を本プログラム例に合わせて変更してください。また、intprg.c と同じく TMU 動作例の 5 ms カウントを使用するため、TMU 動作例を参考に割り込みマスクレベルを追加で変更してください。

## 2.2 使用機能の動作概要

DMAC は DMA 転送要求があると決められたチャンネルの優先順位に従って転送を開始し、転送終了条件が満たされると転送を終了します。転送にはオートリクエスト、外部リクエスト、周辺モジュールリクエストの 3 種類があります。バスモードは、バーストモードとサイクルスチールモードを選択することができます。

表 1 に DMAC の概要を示します。

表 1 DMAC の概要

項目	概要
使用可能チャンネル数	12 チャンネル (チャンネル 0~3 は外部リクエスト受け付け可能)
アドレス空間	4 G バイト
転送データ長	バイト, ワード (2 バイト), ロングワード (4 バイト), 16 バイト, 32 バイト
最大転送回数	16777216 回
アドレスモード	デュアルアドレスモード
転送要求	オートリクエスト, 外部リクエスト, 内蔵周辺モジュールリクエスト
バスモード	サイクルスチールモード, バーストモード
優先順位	チャンネル優先順位固定モード, ラウンドロビンモード
割り込み要求	データ転送ハーフエンド時およびデータ転送終了時, またアドレスエラー発生時に CPU へ割り込み要求を発生可能
外部リクエスト検出	DREQ 入力のロー/ハイレベル検出, 立ち上がり/立ち下がりエッジ検出
転送要求受け付け信号/ 転送終了信号	DACK/DRAK 信号のアクティブレベルをそれぞれ独立に設定可能

2.3 使用機能の設定手順

ここでは、DMAC によるメモリ間転送を行なう場合の初期設定手順について説明します。転送要求はオートリクエストを使用します。

本プログラムでは、初期設定例のプログラムをベースとすることを前提にメイン関数の始めに本設定を行っています。そのため特権モードで動作していることを前提にしていますので、他のプログラムに流用されるなどの場合には処理モードにご注意の上利用してください。

図 1 に LRAM → DDR SDRAM メモリ間転送動作の設定フロー例を示します。各レジスタ設定の詳細は、「SH7780 グループ ハードウェアマニュアル」を参照してください。

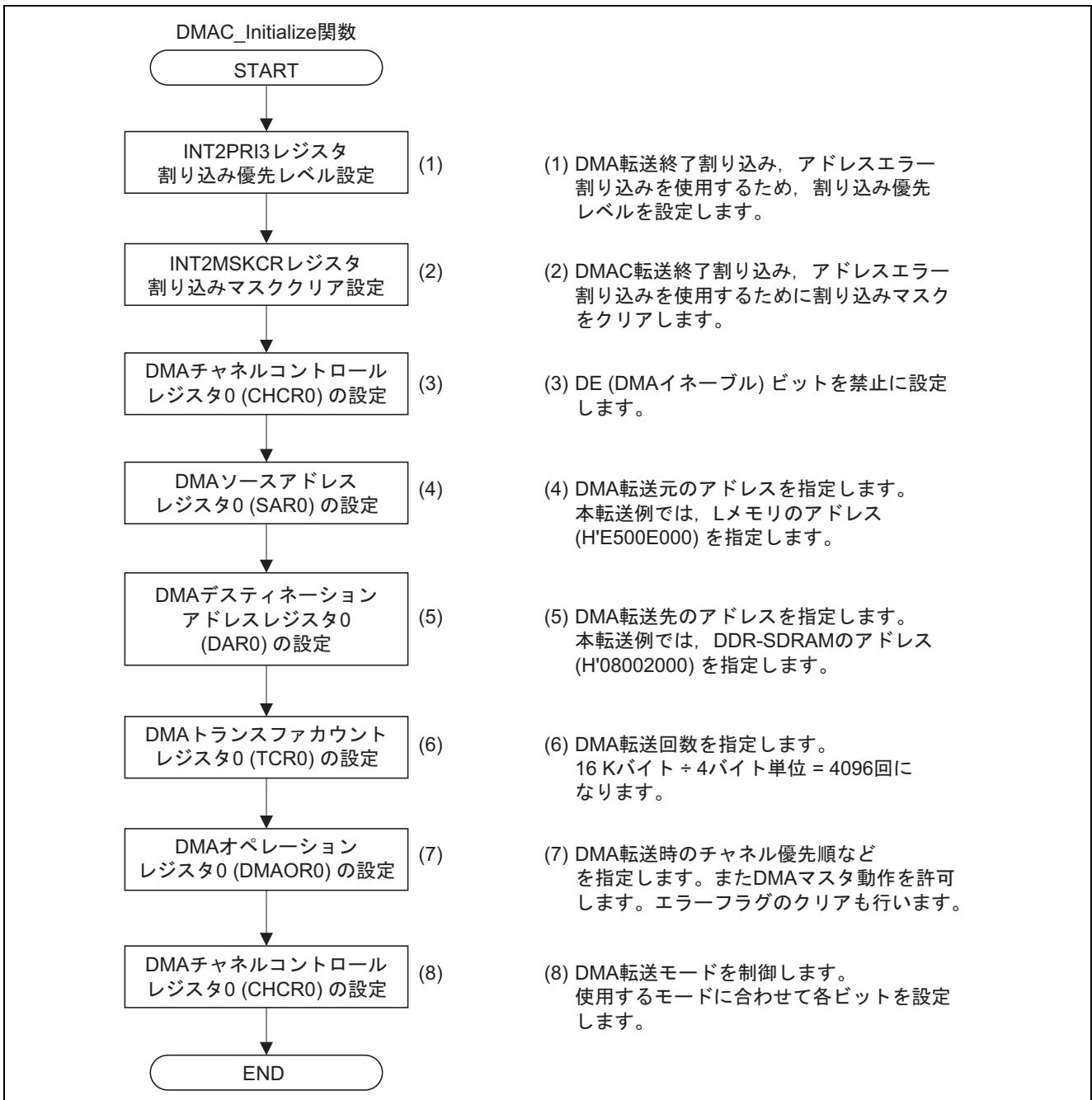


図 1 DMAC メモリ間転送設定フロー例

## 2.4 参考プログラムの処理手順

表 2 に DMAC 関連レジスタの設定例を示します。また、次ページ以降に参考プログラムのメイン関数、転送終了割り込み処理、アドレスエラー割り込み処理、転送再スタート処理のフロー例を示します。

表 2 DMAC の設定

レジスタ名	アドレス	設定値	機能
割り込み優先順位設定レジスタ 3 (INT2PRI3)	H'FFD4000C	H'001F0000	DMAC (0) 割り込み優先レベル: 31
割り込みマスククリアレジスタ (INT2MSKCR)	H'FFD4003C	H'00000100	DMAC (0) 割り込みマスククリア
DMA ソースアドレスレジスタ 0 (SAR0)	H'FC808020	H'E500E000	(転送元アドレス)
DMA デスティネーションアドレスレジスタ 0 (DAR0)	H'FC808024	H'08002000	(転送先アドレス)
DMA トランスファカウントレジスタ 0 (TCR0)	H'FC808028	H'1000	16 K バイト ÷ 4 バイト = 4096 回
DMA チャネルコントロールレジスタ 0 (CHCR0)	H'FC80802C	H'42005414	リピートモード (SAR/DAR/TCR) 4 バイト (ロングワード) 単位転送 転送先アドレス増加 転送元アドレス増加 オートリクエストモード サイクルスチールモード DMA ch0 割り込み要求許可
DMA オペレーションレジスタ 0 (DMAOR0)	H'FC808060	H'3001	インタミットモード 64 DMA マスタイネーブル



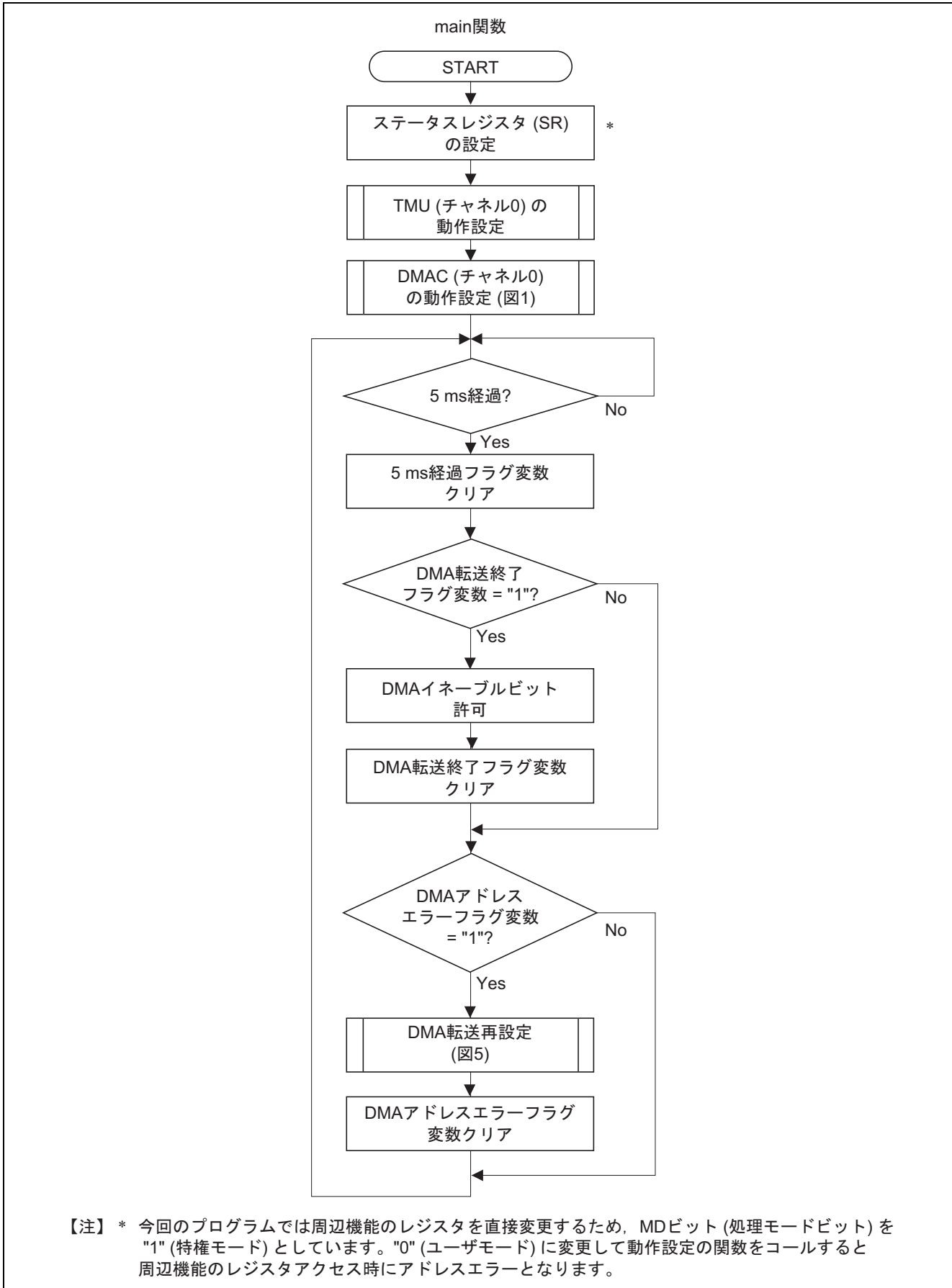


図2 参考プログラム (DMACによるメモリ間転送) メイン関数の処理フロー例

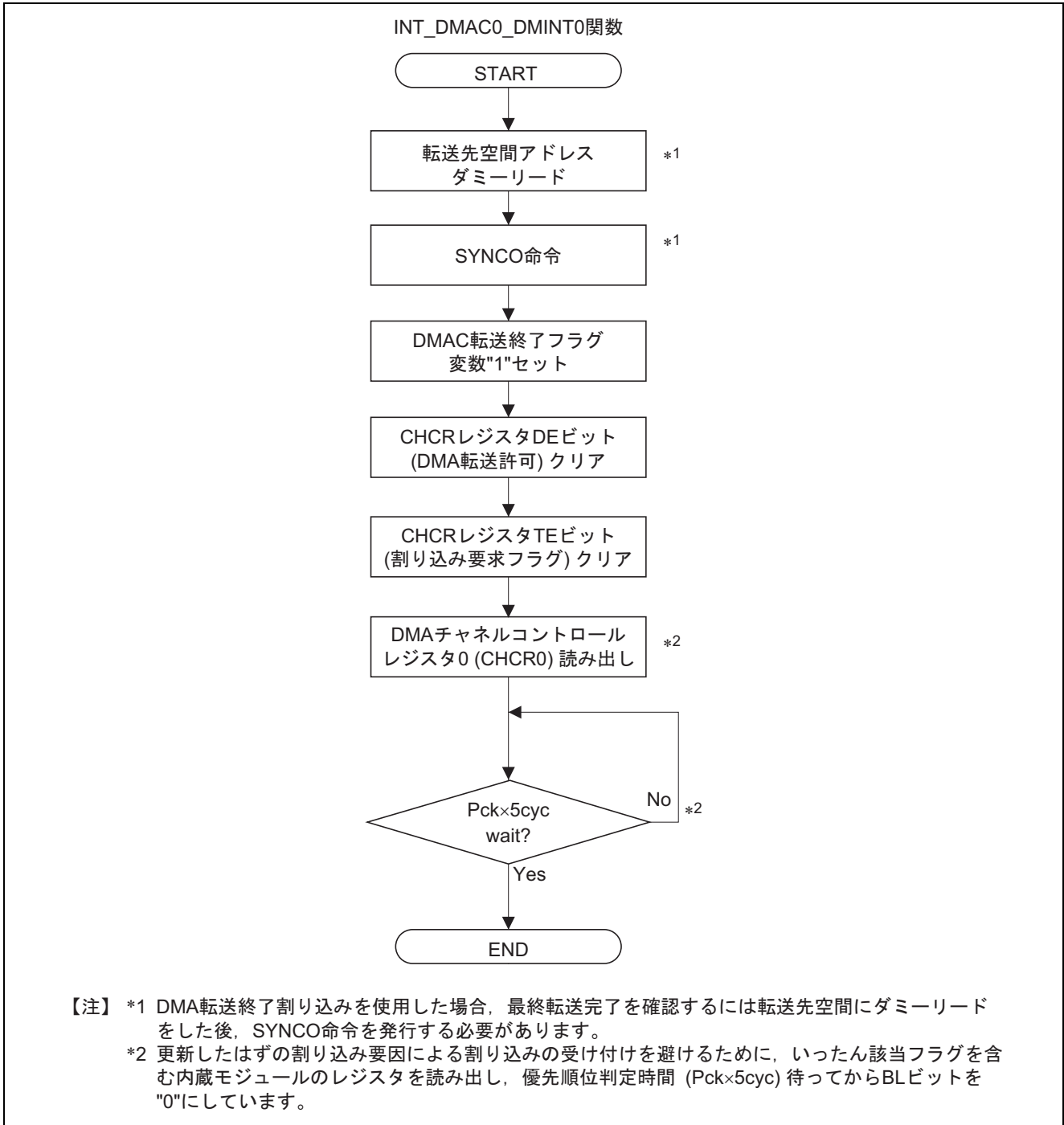


図3 DMINT0 (DMA 転送終了) 割り込み処理フロー例

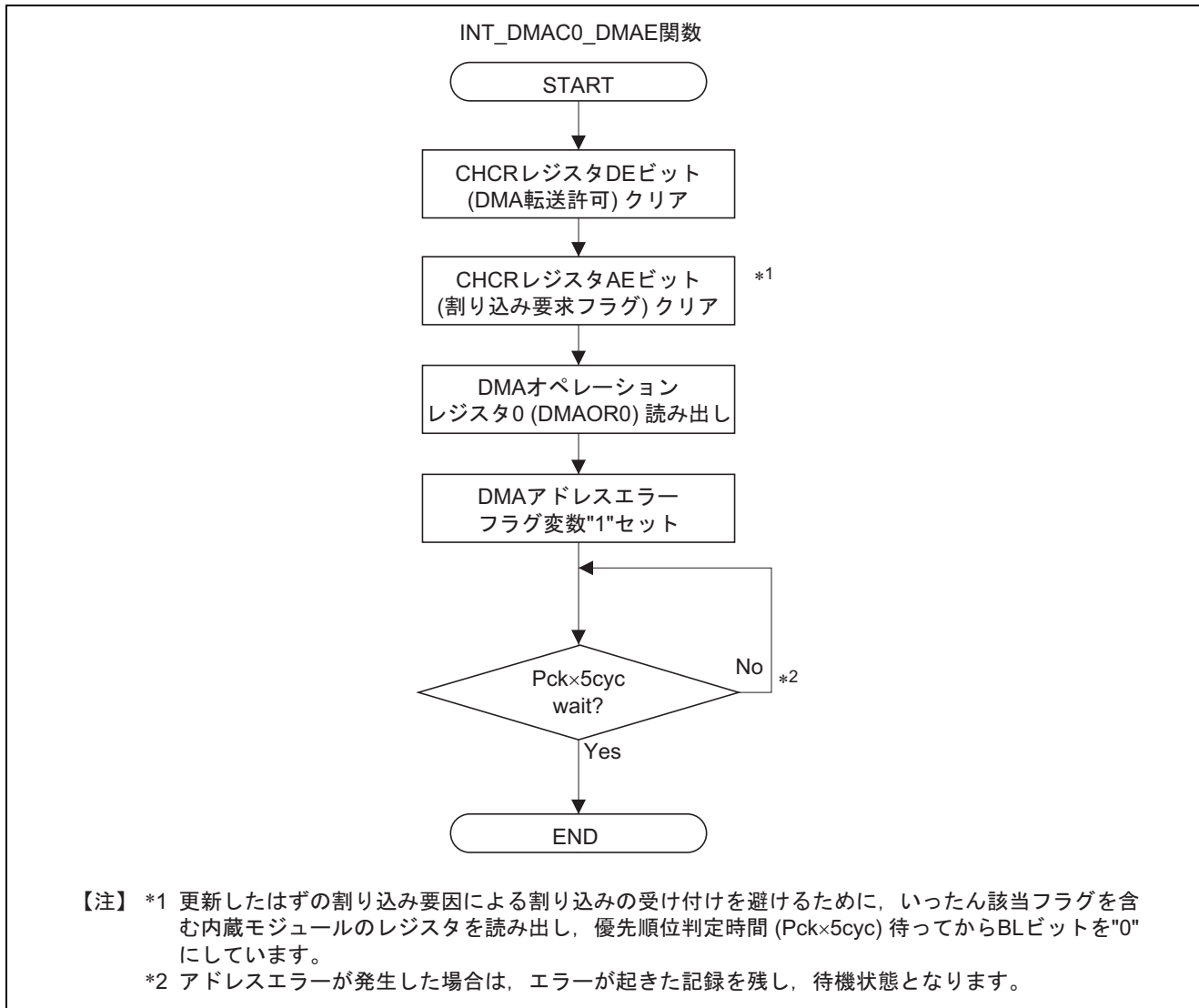


図4 アドレスエラー割り込み処理フロー例

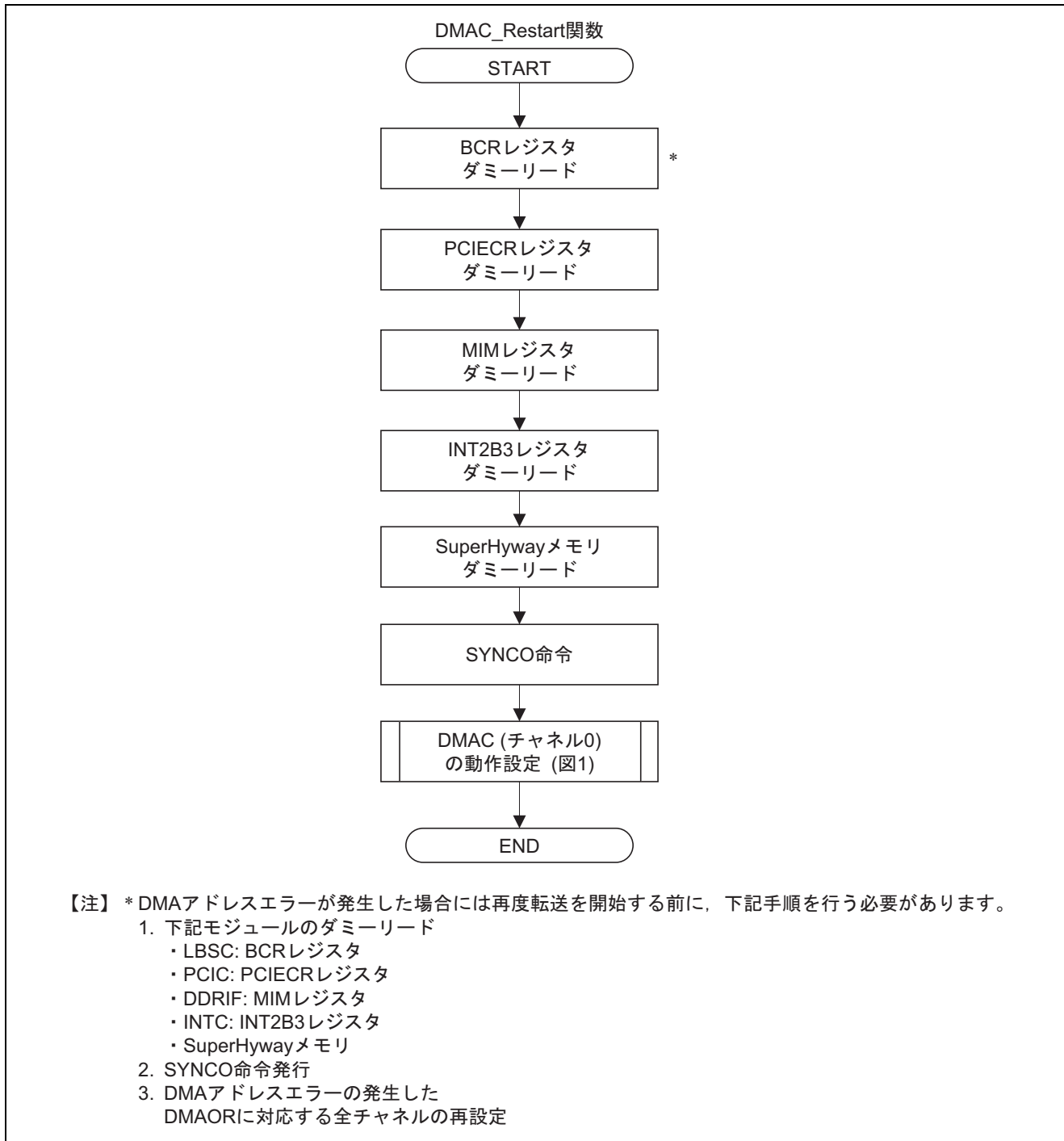


図5 DMA 転送再スタート処理フロー例

### 3. 参考プログラムリスト

#### 1. サンプルプログラムリスト"dmac.c" (1)

```

1  /*"FILE COMMENT"*****
2  *      System Name :   SH7780 Sample Program
3  *      File Name   :   dmac.c
4  *      Version    :   1.00.00
5  *      Contents   :   SH7780 DMAC transmit Program
6  *      Model      :   Hitachi_ULSI_Systems SolutionEngine MS7780SE03
7  *      CPU        :   SH7780
8  *      Compiler   :   SHC.9.1.00
9  *      OS         :   none
10 *
11 *      note       :   < Caution >
12 *                  This sample program is provided simply as a reference and
13 *                  its operation is not guaranteed.
14 *                  Use this sample program as a technical reference when
15 *                  developing software.
16 *
17 *      Copyright(C) 2007 Renesas Technology Corp. All Rights Reserved
18 *
19 *      History    :   2007/12/26 ver 1.00.00
20 *
21 *****/
22 #include <machine.h>
23
24 /* --- Function Definition(internal) --- */
25 void DMAC_Initialize(void);
26 void DMAC_Restart(void);
27
28 /* --- Symbol Definition --- */
29 struct st_dmac0{
30     void          *SAR;          /* SAR */
31     void          *DAR;          /* DAR */
32     unsigned int  TCR;          /* TCR */
33     union{
34         unsigned int LONG;      /* Long Word Access */
35         struct{
36             unsigned int  :1;    /* Bit Access */
37             unsigned int  LCKN :1; /* LCKN */
38             unsigned int  :2;    /* */
39             unsigned int  RPT  :3; /* RPT */
40             unsigned int  :1;    /* */
41             unsigned int  DO   :1; /* DO */
42             unsigned int  RL   :1; /* RL */
43             unsigned int  :1;    /* */
44             unsigned int  TS2  :1; /* TS2 */
45             unsigned int  HE   :1; /* HE */
46             unsigned int  HIE  :1; /* HIE */
47             unsigned int  AM   :1; /* AM */
48             unsigned int  AL   :1; /* AL */
49             unsigned int  DM   :2; /* DM */
50             unsigned int  SM   :2; /* SM */
51             unsigned int  RS   :4; /* RS */
52             unsigned int  DL   :2; /* DL */
53             unsigned int  TB   :1; /* TB */
54             unsigned int  TS   :2; /* TS */

```

## 2. サンプルプログラムリスト"dmac.c" (2)

```

55         unsigned int IE:1;          /* IE */
56         unsigned int TE:1;          /* TE */
57         unsigned int DE:1;          /* DE */
58     } BIT;
59     } CHCR;
60 };
61
62 union st_dmac{                       /* struct DMAOR */
63     unsigned short WORD;             /* Word Access */
64     struct{                           /* Bit Access */
65         unsigned short :2;           /* */
66         unsigned short CMS :2;       /* CMS */
67         unsigned short :2;           /* */
68         unsigned short PR :2;        /* PR */
69         unsigned short :5;           /* */
70         unsigned short AE :1;        /* AE */
71         unsigned short NMIF :1;      /* NMIF */
72         unsigned short DME :1;       /* BIT */
73     } BIT;
74 };
75
76 #define DMAC0      (*(volatile struct st_dmac0 *)0xFC808020) /* DMAC ch0 register top Address */
77 #define DMAOR0     (*(volatile union st_dmac *)0xFC808060) /* DMAOR0 Address */
78
79 #define BCR        (*(volatile unsigned int *)0xFF801000)
80 #define PCIECR     (*(volatile unsigned int *)0xFE000008)
81 #define MIM_L      (*(volatile unsigned int *)0xFE80000C)
82 #define INT2B3     (*(volatile unsigned int *)0xFFD4004C)
83 #define SHMEM0_TOP_ADD ((void *)0xFE410000)
84
85 #define INT2PRI3   (*(volatile unsigned int *)0xFFD4000C)
86 #define INT2MSKCR (*(volatile unsigned int *)0xFFD4003C) /* INT2MSKCR Address */
87
88 #define LRAM_SRC_ADR ((void *)0xE500E000) /* DMA source Address */
89 #define DDR_DST_ADR ((void *)0x08002000) /* DMA destination Address */
90
91 /***** Function Comment *****/
92 * Outline : DMAC_Initialize
93 *-----
94 * Declaration : void DMAC_Initialize(void)
95 *-----
96 * Functional description:
97 * Initialize DMAC controller and start transmit
98 *-----
99 * Return Value : -
100 * Argument : -
101 *-----
102 * Input : -
103 * Output : -
104 *-----
105 * Notes : -
106 *****/

```

## 3. サンプルプログラムリスト"dmac.c" (3)

```

107 void DMAC_Initialize(void)
108 {
109     INT2PRI3 |= 0x001f0000;          /* DMAC ch0 interrupt level 31 */
110     INT2MSKCR = 0x00000100;        /* DMAC ch0 interrupt mask clear */
111
112     DMAC0.CHCR.BIT.DE = 0;         /* DMAC ch0 transmit disable */
113     DMAC0.SAR = LRAM_SRC_ADR;      /* DMAC ch0 transmit source Address */
114     DMAC0.DAR = DDR_DST_ADR;      /* DMAC ch0 transmit destination Address */
115     DMAC0.TCR = 0x00001000;       /* DMAC ch0 transmit count */
116     DMAC0.CHCR.LONG = 0x42005414; /* DMAC ch0 CHCR set */
117     DMAOR0.BIT.CMS = 0x03;        /* Intermittent mode 64 */
118     DMAOR0.BIT.AE &= 0;          /* clear Address Error flag */
119     DMAOR0.BIT.NMIF &= 0;        /* clear NMI interrupt flag */
120     DMAOR0.BIT.DME = 1;          /* enable DMAC ch0 DMA master */
121     DMAC0.CHCR.BIT.DE = 1;        /* enable DMAC ch0 transmit */
122 }
123
124 /***** Function Comment *****/
125 * Outline      : DMAC_ReStart
126 *-----
127 * Declaration  : void DMAC_Restart(void)
128 *-----
129 * Functional description:
130 *      restart DMA Controller transmit
131 *-----
132 * Return Value : -
133 * Argument     : -
134 *-----
135 * Input        : -
136 * Output       : -
137 *-----
138 * Notes        : -
139 *****/
140 void DMAC_Restart(void)
141 {
142     volatile unsigned int dummy;
143
144     dummy = BCR;          /* dummy read BCR */
145     dummy = PCIECR;      /* dummy read PCIECR */
146     dummy = MIM_L;      /* dummy read MIM */
147     dummy = INT2B3;     /* dummy read INT2B3 */
148     dummy = (*(unsigned int *)SHMEM0_TOP_ADD); /* dummy read SuperHyway RAM */
149
150     synco();            /* SYNCO */
151
152     DMAC_Initialize();
153 }
    
```

## 4. サンプルプログラムリスト"intprg.c" (1)

```

1  /*"FILE COMMENT"*****
2  *      System Name :   SH7780 Sample Program
3  *      File Name   :   intprg.c
4  *      Version    :   1.00.00
5  *      Contents   :   SH7780 Initialize Program
6  *      Model      :   Hitachi_ULSI_Systems SolutionEngine MS7780SE03
7  *      CPU        :   SH7780
8  *      Compiler   :   SHC.9.1.00
9  *      OS         :   none
10 *
11 *      note       :   < Caution >
12 *                  This sample program is provided simply as a reference and
13 *                  its operation is not guaranteed.
14 *                  Use this sample program as a technical reference when
15 *                  developing software.
16 *
17 * Copyright(C) 2007 Renesas Technology Corp. All Rights Reserved
18 *
19 *      History    :   2007/12/26 ver 1.00.00
20 *
21 *****/
22 #include <machine.h>
23
24 /* --- Function Definition(internal) --- */
25 static void int_responstime_wait(unsigned int wait_time);
26
27 /* --- Symbol Definition --- */
28 struct st_tmu{                               /* struct TMU0 */
29     unsigned int      TCOR;                   /* TCOR      */
30     unsigned int      TCNT;                   /* TCNT      */
31     union {
32         unsigned short WORD;                 /* Word Access */
33         struct {
34             unsigned short      :7; /* Bit Access */
35             unsigned short UNF   :1; /* UNF      */
36             unsigned short      :2; /*          */
37             unsigned short UNIE  :1; /* UNIE     */
38             unsigned short CKEG  :2; /* CKEG     */
39             unsigned short TPSC  :3; /* TPSC     */
40         } BIT;
41     } TCR;
42 };
43
44 union st_int2b3{
45     unsigned int      LONG; /* Long Word Access */
46     struct{
47         unsigned int      :18; /* Bit Access */
48         unsigned int DMAE1 :1; /* DMAE1 */
49         unsigned int DMAE0 :1; /* DMAE0 */
50         unsigned int DMINT11 :1; /* DMINT11 */
51         unsigned int DMINT10 :1; /* DMINT10 */
52         unsigned int DMINT9  :1; /* DMINT9 */
53         unsigned int DMINT8  :1; /* DMINT8 */
54         unsigned int DMINT7  :1; /* DMINT7 */
55         unsigned int DMINT6  :1; /* DMINT6 */
56         unsigned int DMINT5  :1; /* DMINT5 */

```



## 5. サンプルプログラムリスト"intprg.c" (2)

```

57     unsigned int    DMINT4 :1;    /* DMINT4 */
58     unsigned int    DMINT3 :1;    /* DMINT3 */
59     unsigned int    DMINT2 :1;    /* DMINT2 */
60     unsigned int    DMINT1 :1;    /* DMINT1 */
61     unsigned int    DMINT0 :1;    /* DMINT0 */
62     }BIT;
63 };
64
65 struct st_dmac0{                /* struct DMAC0 */
66     void            *SAR;        /* SAR */
67     void            *DAR;        /* DAR */
68     unsigned int    TCR;        /* TCR */
69     union{                /* CHCR */
70         unsigned int LONG;      /* Long Word Access */
71         struct{                /* Bit Access */
72             unsigned int :1;    /* */
73             unsigned int LCKN :1; /* LCKN */
74             unsigned int :2;    /* */
75             unsigned int RPT :3; /* RPT */
76             unsigned int :1;    /* */
77             unsigned int DO :1; /* DO */
78             unsigned int RL :1; /* RL */
79             unsigned int :1;    /* */
80             unsigned int TS2 :1; /* TS2 */
81             unsigned int HE :1; /* HE */
82             unsigned int HIE :1; /* HIE */
83             unsigned int AM :1; /* AM */
84             unsigned int AL :1; /* AL */
85             unsigned int DM :2; /* DM */
86             unsigned int SM :2; /* SM */
87             unsigned int RS :4; /* RS */
88             unsigned int DL :2; /* DL */
89             unsigned int TB :1; /* TB */
90             unsigned int TS :2; /* TS */
91             unsigned int IE :1; /* IE */
92             unsigned int TE :1; /* TE */
93             unsigned int DE :1; /* DE */
94         } BIT;
95     } CHCR;
96 };
97
98 union st_dmac{                /* struct DMAOR */
99     unsigned short WORD;        /* Word Access */
100    struct{                /* Bit Access */
101        unsigned short :2;    /* */
102        unsigned short CMS :2; /* CMS */
103        unsigned short :2;    /* */
104        unsigned short PR :2; /* PR */
105        unsigned short :5;    /* */
106        unsigned short AE :1; /* AE */
107        unsigned short NMIF :1; /* NMIF */
108        unsigned short DME :1; /* BIT */
109    } BIT;
110 };
111

```

## 6. サンプルプログラムリスト"intprg.c" (3)

```

112 #define DMAC0      (*(volatile struct st_dmac0 *)0xFC808020) /* DMAC ch0 register top Address */
113 #define DMAOR0     (*(volatile union st_dmac *) 0xFC808060) /* DMAOR0 Address */
114
115 #define INT2B3     (*(volatile union st_int2b3 *)0xFFD4004C) /* INT2B3 Address */
116
117 #define TMU0       (*(volatile struct st_tmu *) 0xFFD80008) /* TMU0 Address */
118 #define DDR_TOP_ADR (*(volatile unsigned int *) 0x08000000)
119
120 #define INTC_RESPONSEWAIT (0x00000014) /* INT response wait Pck 5cycle
121                                     H'14 = (1/Pck*5cyc)/(1/Ick*3cyc) */
122
123 /* --- RAM allocation variable declaration --- */
124 extern unsigned char ulFlg5ms;
125 extern unsigned char ulFlgEndDmac0;
126 extern unsigned char ulFlgAddErrDMAC0;
127 #pragma section IntPRG

293 /* H'580 TMU ch-0 underflow interrupt */
294 void INT_TMU0_TUNI0(void)
295 {
296     volatile unsigned short dummy;
297
298     TMU0.TCR.BIT.UNF = 0; /* TMU ch0 UNF flag clear */
299     ulFlg5ms = 1; /* set 5ms flag */
300     dummy = TMU0.TCR.WORD; /* dummy read */
301
302     int_responstime_wait(INTC_RESPONSEWAIT); /* 5cyc(Pck=33MHz) wait */
303 }

321 /* H'640 ch-0 DMA transmit end or halfend interrupt */
322 void INT_DMAC0_DMINT0(void)
323 { /* (start of additional part from Initialize program) */
324     volatile unsigned int dummy;
325
326     dummy = DDR_TOP_ADR; /* destinate address dummy read */
327     synco(); /* SYNCO */
328
329     ulFlgEndDmac0 = 1; /* set "DMAC ch0 transmit end flag" */
330
331     DMAC0.CHCR.BIT.DE = 0; /* DMAC ch0 transmit disable */
332     DMAC0.CHCR.BIT.TE &= 0; /* DMAC ch0 TEBit clear */
333
334     dummy = DMAC0.CHCR.LONG; /* dummy read */
335
336     int_responstime_wait(INTC_RESPONSEWAIT); /* 5cyc(Pck=33MHz) wait */
337 } /* (end of additional part from Initialize program) */
338 /* H'660 ch-1 DMA transmit end or halfend interrupt */
339 void INT_DMAC0_DMINT1(void)
340 {
341 }
342 /* H'680 ch-2 DMA transmit end or halfend interrupt */
343 void INT_DMAC0_DMINT2(void)
344 {
345 }
346

```

## 7. サンプルプログラムリスト"intprg.c" (4)

```

347  /* H'6A0 ch-3 DMA transmit end or halfend interrupt */
348  void INT_DMACH0_DMINT3(void)
349  {
350  }
351  /* H'6C0 ch0-5,ch6-11 DMA address error interrupt */
352  void INT_DMACH0_DMAE(void)
353  {
354      volatile unsigned int dummy;
355
356      DMACH0.CHCR.BIT.DE = 0;          /* DMAC ch0 transmit disable */
357      DMAOR0.BIT.AE &= 0;             /* clear address error flag */
358
359      dummy = DMAOR0.WORD;             /* dummy read */
360
361      ulFlgAddErrDMACH0 = 1;           /* set "DMAC ch0-5 address error flag" */
362
363      int_responstime_wait(INTC_RESPONSEWAIT); /* 5cyc(Pck=33MHz) wait */
364  }
365
366  #pragma inline_asm(int_responstime_wait)
367  static void int_responstime_wait(unsigned int wait_time)
368  {
369      ?0001:
370          DT          R4
371          BF          ?0001
372          NOP
373  }

```

## 8. サンプルプログラムリスト"main.c" (1)

```

1  /*"FILE COMMENT"*****
2  *      System Name :   SH7780 Sample Program
3  *      File Name   :   main.c
4  *      Version    :   1.00.00
5  *      Contents   :   SH7780 Initialize Program
6  *      Model      :   Hitachi_ULSI_Systems SolutionEngine MS7780SE03
7  *      CPU        :   SH7780
8  *      Compiler   :   SHC.9.1.00
9  *      OS         :   none
10 *
11 *      note       :   < Caution >
12 *                  This sample program is provided simply as a reference and
13 *                  its operation is not guaranteed.
14 *                  Use this sample program as a technical reference when
15 *                  developing software.
16 *
17 * Copyright(C) 2007 Renesas Technology Corp. All Rights Reserved
18 *
19 *      History    :   2007/12/26 ver 1.00.00
20 *
21 *****/
22 #include <machine.h>
23
24 /* --- Function Definition(internal) --- */
25
26 /* --- Symbol Definition --- */
27 struct st_dmac0{                               /* struct DMAC0 */
28     void          *SAR;                         /* SAR */
29     void          *DAR;                         /* DAR */
30     unsigned int  TCR;                         /* TCR */
31     union{                                       /* CHCR */
32         unsigned int LONG;                     /* Long Word Access */
33         struct{
34             unsigned int      :1;             /*
35             unsigned int LCKN :1;             /* LCKN
36             unsigned int      :2;             /*
37             unsigned int RPT  :3;             /* RPT
38             unsigned int      :1;             /*
39             unsigned int DO   :1;             /* DO
40             unsigned int RL   :1;             /* RL
41             unsigned int      :1;             /*
42             unsigned int TS2  :1;             /* TS2
43             unsigned int HE   :1;             /* HE
44             unsigned int HIE  :1;             /* HIE
45             unsigned int AM   :1;             /* AM
46             unsigned int AL   :1;             /* AL
47             unsigned int DM   :2;             /* DM
48             unsigned int SM   :2;             /* SM
49             unsigned int RS   :4;             /* RS
50             unsigned int DL   :2;             /* DL
51             unsigned int TB   :1;             /* TB
52             unsigned int TS   :2;             /* TS
53             unsigned int IE   :1;             /* IE
54             unsigned int TE   :1;             /* TE
55             unsigned int DE   :1;             /* DE
56         } BIT;
    
```

## 9. サンプルプログラムリスト"main.c" (2)

```

57         } CHCR;
58     };
59
60     union st_dmac{                               /* struct DMAOR */
61         unsigned short WORD;                     /* Word Access */
62         struct{                                   /* Bit Access */
63             unsigned short      :2;              /*          */
64             unsigned short CMS   :2;              /*    CMS    */
65             unsigned short      :2;              /*          */
66             unsigned short PR    :2;              /*    PR     */
67             unsigned short      :5;              /*          */
68             unsigned short AE    :1;              /*    AE     */
69             unsigned short NMIF   :1;              /*   NMIF    */
70             unsigned short DME   :1;              /*    BIT    */
71         } BIT;
72     };
73
74     #define DMAC0 (*(volatile struct st_dmac0 *) 0xFC808020) /* DMAC ch0 register top Address */
75
76     #define SR_Init 0x400000e0 /* Privileged mode, IMASK level 14 */
77
78     /* --- RAM allocation variable declaration --- */
79     volatile unsigned char ulFlg5ms;
80     volatile unsigned char ulCnt1s;
81     volatile unsigned char ulFlgEndDmac0;
82     volatile unsigned char ulFlgAddErrDMAC0;
83
84     /***** Function Comment *****/
85     * Outline      : main
86     *-----
87     * Declaration  : void main(void)
88     *-----
89     * Functional description:
90     *      main function
91     *-----
92     * Return Value : -
93     * Argument     : -
94     *-----
95     * Input       : -
96     * Output      : -
97     *-----
98     * Notes       : -
99     /***** Function Comment End *****/
100 void main(void)
101 {
102     set_cr(SR_Init);                               /* Set SR "Privileged mode, IMASK level 14" */
103
104     TMU0_Initialize();                             /* TMU0 Initialize (additional part from Initialize program)*/
105     DMAC_Initialize();                             /* DMAC Initialize (additional part from Initialize program)*/
106
107     while(1)
108     {                                               /* (start of additional part from Initialize program) */
109         while(ulFlg5ms == 0x00)
110         {
111             nop();

```

## 10. サンプルプログラムリスト"main.c" (3)

```

112     }
113     ulFlg5ms = 0;           /* clear 5ms flag */
114     ulCnt1s++;           /* 1s count +1 */
115
116     if(ulFlgEndDmac0 == 1) /* if DMAC-ch0 transmit end */
117     {
118         DMAC0.CHCR.BIT.DE = 1; /* DMAC ch0 transmit enable */
119         ulFlgEndDmac0 = 0; /* clear "DMAC ch0 transmit end flag" */
120     }
121     if(ulFlgAddErrDMAC0 == 1)
122     {
123         DMAC_Restart(); /* DMAC-ch0 transmit restart */
124         ulFlgAddErrDMAC0 = 0;
125     }
126 } /* (end of additional part from Initialize program) */
127 }
    
```

## 11. サンプルプログラムリスト"vecttbl.src"

```

1  ;*"FILE COMMENT"*****
2  ;      System Name :   SH7780 Sample Program
3  ;      File Name   :   vecttbl.src
4  ;      Version    :   1.00.00
5  ;      Contents   :   SH7780 Initialize Program
6  ;      Model      :   Hitachi_ULSI_Systems SolutionEngine MS7780SE03
7  ;      CPU        :   SH7780
8  ;      Compiler   :   SHC.9.1.00
9  ;      OS         :   none
10 ;
11 ;      note       :   < Caution >
12 ;                  This sample program is provided simply as a reference and
13 ;                  its operation is not guaranteed.
14 ;                  Use this sample program as a technical reference when
15 ;                  developing software.
16 ;
17 ; Copyright (C) 2007 Renesas Technology Corp. All Rights Reserved
18 ;
19 ;      History    :   2007/12/26 ver 1.00.00
20 ;
21 ;*****/
22
23      .include    "vect.inc"
24
25      .section    VECTTBL,data
26      .export     _RESET_VECTORS

305 ;TMU-ch0
306      ;H'580      TMU_TUNIO
307      .data.b     H'F0

320 ;DMAC(0)
321      ;H'640      DMINT0
322      .data.b     H'F0      /* (change part from Initialize program) */
323      ;H'660      DMINT1
324      .data.b     H'00
325      ;H'680      DMINT2
326      .data.b     H'00
327      ;H'6A0      DMINT3
328      .data.b     H'00
329      ;H'6C0      DMAE
330      .data.b     H'F0      /* (change part from Initialize program) */
331      ;H'6E0      reserve
332      .data.b     H'00
    
```

#### 4. 参考ドキュメント

- ハードウェアマニュアル  
SH7780 ハードウェアマニュアル  
(最新版をルネサス テクノロジホームページから入手してください。)
- ソフトウェアマニュアル  
SH-4A ソフトウェアマニュアル  
(最新版をルネサス テクノロジホームページから入手してください。)



ホームページとサポート窓口

ルネサス テクノロジホームページ

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

お問合せ先

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

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

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2008.03.21	—	初版発行

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。

### 本資料ご利用に際しての留意事項

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

D039444