

## SH7231 グループ

R01AN1264JJ0100

Rev.1.00

2012.12.03

## DMAC を用いた内蔵周辺モジュールへのデータ転送例

### 要旨

本アプリケーションノートでは、SH7231のダイレクトメモリアクセスコントローラ (DMAC) を用いた内蔵周辺モジュールへのデータ転送について説明します。

動作の特長を以下に示します。

- DMAC のチャンネル 1 を使用しています。
- DMA 転送の起動要因は、シリアルコミュニケーションインタフェース (SCI チャンネル 0) の送信データエンプティです。
- バスモードはサイクルスチールモードです。
- 転送元は内蔵 RAM、転送先は SCI チャンネル 0 のトランスミットデータレジスタです。内蔵 RAM に置かれた文字列データを SCI から送信します。

### 対象デバイス

SH7231

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

## 目次

|                                      |    |
|--------------------------------------|----|
| 1. 仕様 .....                          | 3  |
| 2. 動作確認条件 .....                      | 4  |
| 3. 関連アプリケーションノート .....               | 4  |
| 4. 周辺機能説明 .....                      | 5  |
| 4.1 ダイレクトメモリアクセスコントローラ (DMAC) .....  | 5  |
| 4.2 シリアルコミュニケーションインタフェース (SCI) ..... | 8  |
| 5. ハードウェアウェア説明 .....                 | 11 |
| 5.1 使用端子一覧 .....                     | 11 |
| 6. ソフトウェア説明 .....                    | 12 |
| 6.1 動作概要 .....                       | 12 |
| 6.2 ファイル構成 .....                     | 14 |
| 6.3 構造体/列挙体一覧 .....                  | 15 |
| 6.4 変数一覧 .....                       | 16 |
| 6.5 関数一覧 .....                       | 16 |
| 6.6 関数仕様 .....                       | 17 |
| 6.7 フローチャート .....                    | 19 |
| 7. 参考ドキュメント .....                    | 23 |

## 1. 仕様

本アプリケーションノートでは、ダイレクトメモリアクセスコントローラ(DMAC)を使用して、内蔵 RAM からシリアルコミュニケーションインタフェース (SCI チャンネル 0) へのデータ転送を行います。

サンプルコードでは、内蔵 RAM に格納した文字列データを DMA 転送により SCI チャンネル 0 から送信しています。DMAC をサイクルスチールモードに設定し、DMA 転送の起動要因には SCI チャンネル 0 の送信データエンプティを使用します。転送データサイズはバイト転送で、転送数は転送元の文字列の数だけとしています。また、1 回の転送要求に対する転送回数は 1 回としています。

表 1.1 に使用する周辺機能と用途を、図 1.1 に使用する周辺機能のブロック図を示します。

表 1.1 使用する周辺機能と用途

| 周辺機能                               | 用途          |
|------------------------------------|-------------|
| ダイレクトメモリアクセスコントローラ (DMAC)          | データの DMA 転送 |
| 内蔵 RAM                             | 転送元         |
| シリアルコミュニケーションインタフェース (SCI チャンネル 0) | 転送先         |

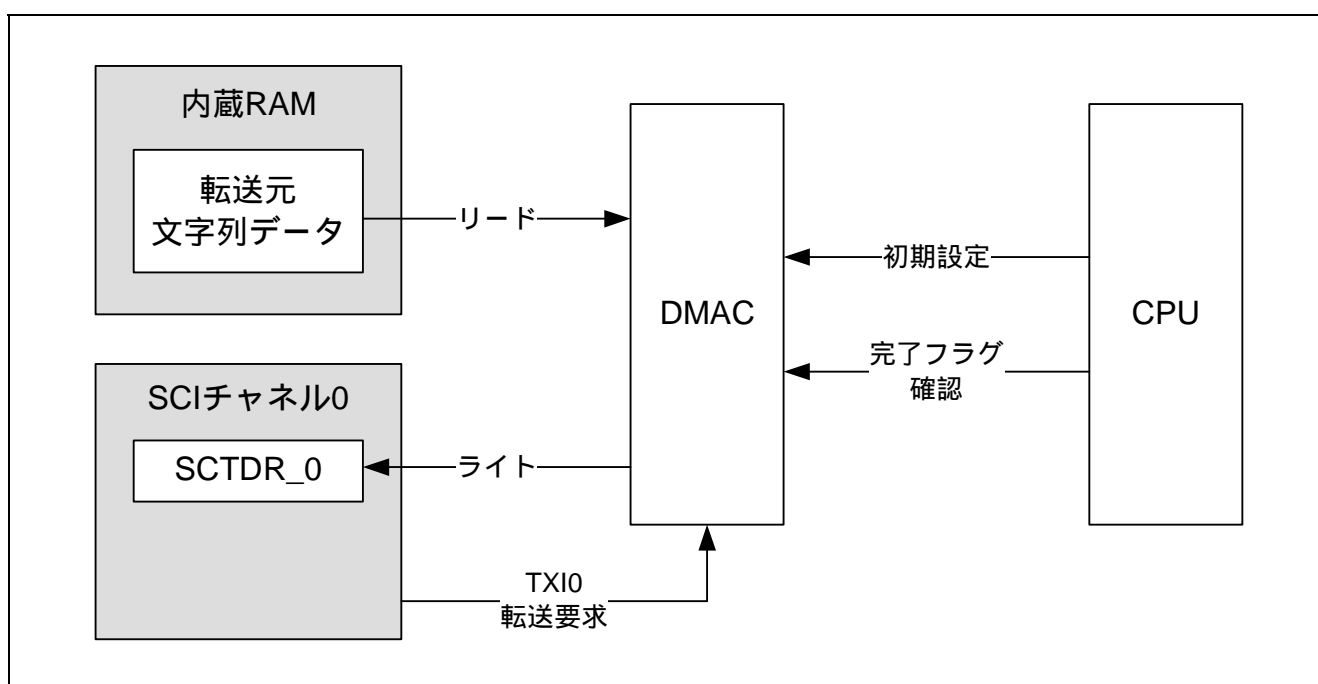


図 1.1 使用する周辺機能のブロック図

## 2. 動作確認条件

本アプリケーションノートのサンプルコードは、下記の条件で動作を確認しています。

表2.1 動作確認条件

| 項目            | 内容  |
|---------------|---|
| 使用マイコン        | SH7231  |
| 動作周波数         | メインクロック : 100MHz<br>バスクロック : 50MHz<br>周辺クロック : 50MHz  |
| 動作電圧          | Vcc : 3.3V  |
| 統合開発環境        | ルネサス エレクトロニクス製<br>High-performance Embedded Workshop Ver.4.08.00  |
| C コンパイラ       | ルネサス エレクトロニクス製 SuperH RISC engine ファミリ<br>C/C++コンパイラパッケージ Ver.9.04 Release 00<br>コンパイルオプション<br>-cpu=sh2afpu -fpu=single<br>-include="\$(WORKSPDIR)¥inc"<br>-object="\$(CONFIGDIR)¥\$(FILELEAF).obj" -debug -gbr=auto -chgincpath<br>-errorpath -global_volatile=0 -opt_range=all -infinite_loop=0<br>-del_vacant_loop=0 -struct_alloc=1 -nologo |
| 動作モード         | シングルチップモード  |
| サンプルコードのバージョン | 1.00  |
| 使用ボード         | R0K572310C000BR   |

## 3. 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。併せて参照してください。

- SH7231 グループ 初期設定例 (R01AN0322JJ)

## 4. 周辺機能説明

### 4.1 ダイレクトメモリアクセスコントローラ (DMAC)

ダイレクトメモリアクセスコントローラ (DMAC) について説明します。基本的な内容は「SH7231 グループ ユーザーズマニュアル ハードウェア編」に記載しています。

DMAC は DMA 転送があると、決められたチャンネルの優先順位にしたがって転送を開始し、転送終了条件が満たされると転送を終了します。転送要求にはオートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストの 3 種類のモードがあります。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から選択することができます。

表 4.1 に DMAC の概要を示します。図 4.1 にサイクルスチール通常モードの DMA 転送例を、図 4.2 にバーストモードの DMA 転送例を示します。また、図 4.3 に DMAC のブロック図を示します。

表 4.1 DMAC の概要

| 項目                      | 内容  |
|-------------------------|---|
| チャンネル数                  | CH0~CH3 の 4 チャンネル<br>(CH0~CH1 の 2 チャンネルのみ、外部リクエストの受け付けが可能)  |
| アドレス空間                  | アーキテクチャ上は 4 GB  |
| 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)、<br>16 バイト (ロングワード × 4)  |
| 最大転送回数                  | 16,777,216 (24 ビット) 回   |
| アドレスモード                 | シングルアドレスモード、デュアルアドレスモード   |
| 転送要求                    | 外部リクエスト、内蔵周辺モジュールリクエスト、オートリクエスト<br>(SCI: 8 要因、SCIF: 8 要因、IIC3: 2 要因、LVDS: 1 要因*、A/D 変換器: 2 要因、<br>MTU2S: 2 要因、MTU2: 5 要因、CMT2: 5 要因、CMT: 2 要因、RSPI: 2 要因、<br>RCAN-ET: 1 要因) |
| バスモード                   | サイクルスチールモード (通常モード、インターミットモード)、<br>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 1/2 終了時またはデータ転送終了時に CPU へ割り込み要求発生   |
| 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<br>信号 / 転送終了信号 | DACK および TEND はアクティブレベルを設定可能  |

【注】 \* 転送要求に内蔵周辺モジュールリクエストの LVDS 要因があるのは SH72315A のみ

サイクルスチールの通常モードでは、DMAC は一回の転送単位（バイト、ワード、ロングワード、または 16バイト単位）の転送を終了するたびにバス権を他のバスマスタに渡します。その後転送要求があれば、他のバスマスタからバス権を取り戻し、再び1転送単位の転送を行い、その転送を終了するとまたバス権を他のバスマスタに渡します。これを転送終了条件が満たされるまで繰り返します。  
 サイクルスチール通常モードは、転送要求元、転送元、転送先にかかわらずすべての転送区間で使えます。

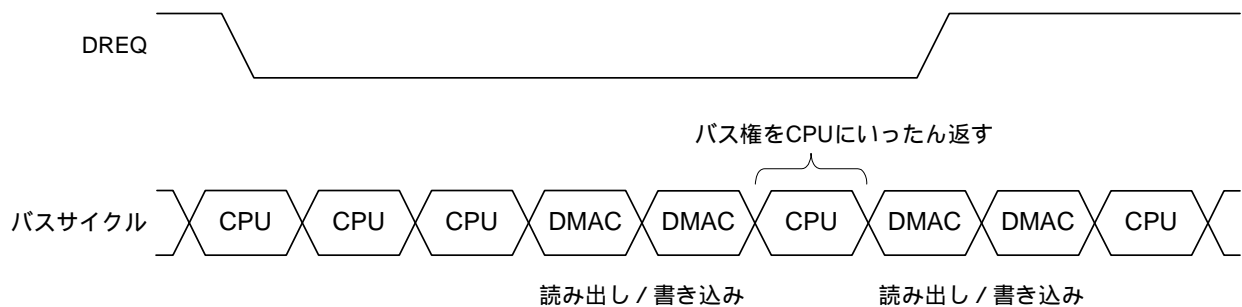


図4.1 サイクルスチール通常モードの DMA 転送例（デュアルアドレス、DREQ ローレベル検出）

バーストモードでは、DMACは一度バス権を取ると、転送終了条件が満たされるまでバス権を解放せずに転送を続けます。ただし、外部リクエストモードで、DREQをレベルで検出する場合には、DREQがアクティブなレベルでなくなると、転送終了条件が満たされていなくても、すでに要求を受け付けたDMAC転送要求を終了後に他のバスマスタにバス権を渡します。

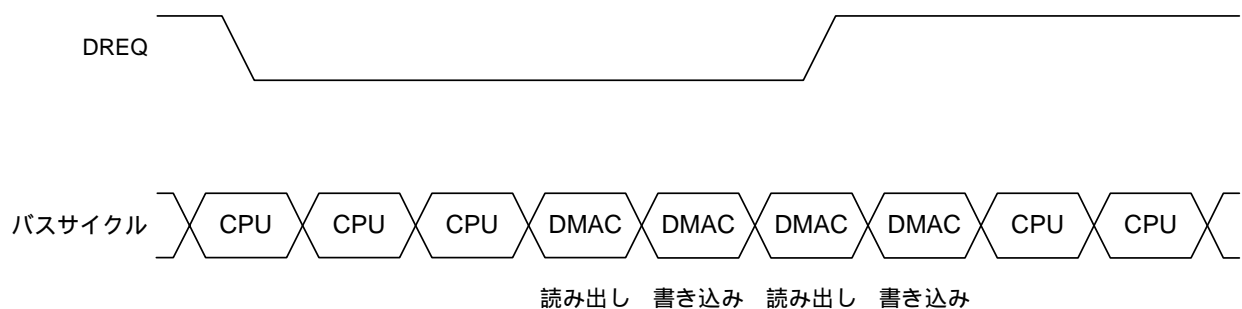


図4.2 バーストモードの DMA 転送例（デュアルアドレス、DREQ ローレベル検出）

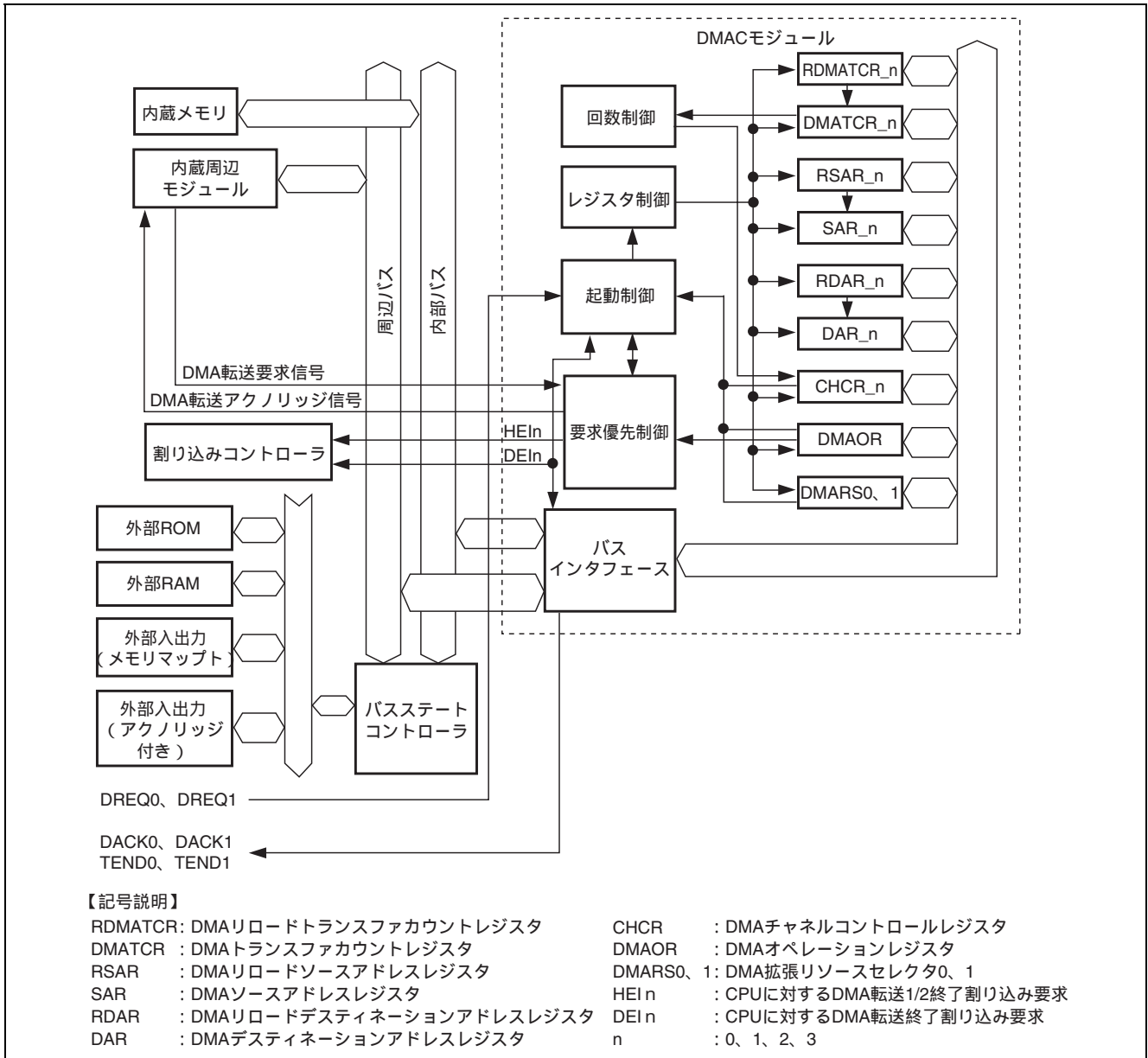


図4.3 DMAC のブロック図

## 4.2 シリアルコミュニケーションインタフェース (SCI)

シリアルコミュニケーションインタフェース(SCI)について説明します。基本的な内容は「SH7231 グループ ユーザーズマニュアル ハードウェア編」に記載しています。

SCI は調歩同期式通信とクロック同期式通信の 2 方式をサポートします。調歩同期式では、シリアルデータ通信フォーマットを 12 種類から選択できます。また、調歩同期式では複数のプロセッサ間の通信機能(マルチプロセッサ通信機能)を備えています。

表 4.2にSCI の概要を示します。図 4.4に調歩同期式通信のデータフォーマット例を、図 4.5にクロック同期式通信のデータフォーマット例を示します。また、図 4.6にSCI のブロック図を示します。

表4.2 SCI の概要

| 項目             | 内容  |
|----------------|---|
| チャンネル数         | チャンネル 0~3 の 4 チャンネル   |
| 通信モード          | 調歩同期式シリアル通信、クロック同期式シリアル通信<br>全二重通信可能  |
| 調歩同期式の通信フォーマット | データ長：7 ビットまたは 8 ビット<br>ストップビット長：1 ビットまたは 2 ビット<br>パリティ：偶数パリティ、奇数パリティ、またはパリティなし          |
| クロックソース        | ボーレートジェネレータ(内部クロック)、または SCK 端子入力(外部クロック)  |
| 割り込み           | 送信データエンプティ割り込み、送信終了割り込み、<br>受信データフル割り込み、および受信エラー割り込み                                    |
| その他            | マルチプロセッサ間通信が可能<br>受信エラーを検出可能<br>ブレークを検出可能<br>LSB ファースト/MSB ファーストを選択可能(調歩同期 7 ビットデータを除く) |



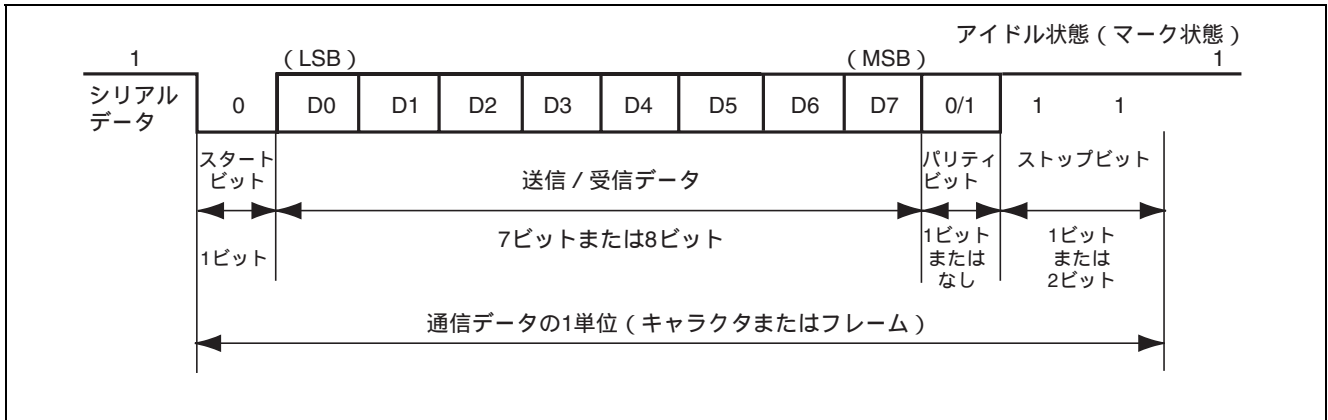


図4.4 調歩同期式通信のデータフォーマット例

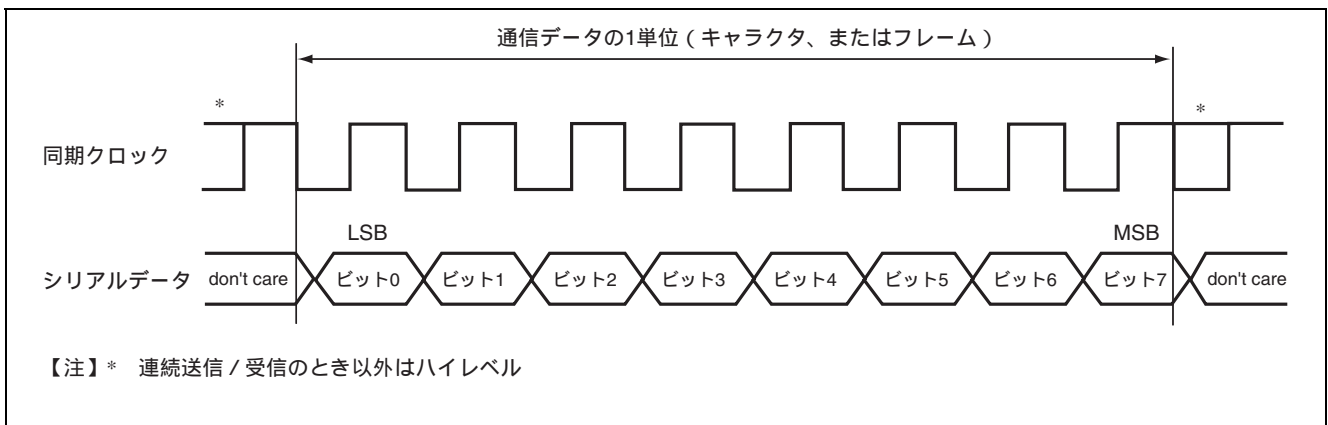


図4.5 クロック同期式通信のデータフォーマット例

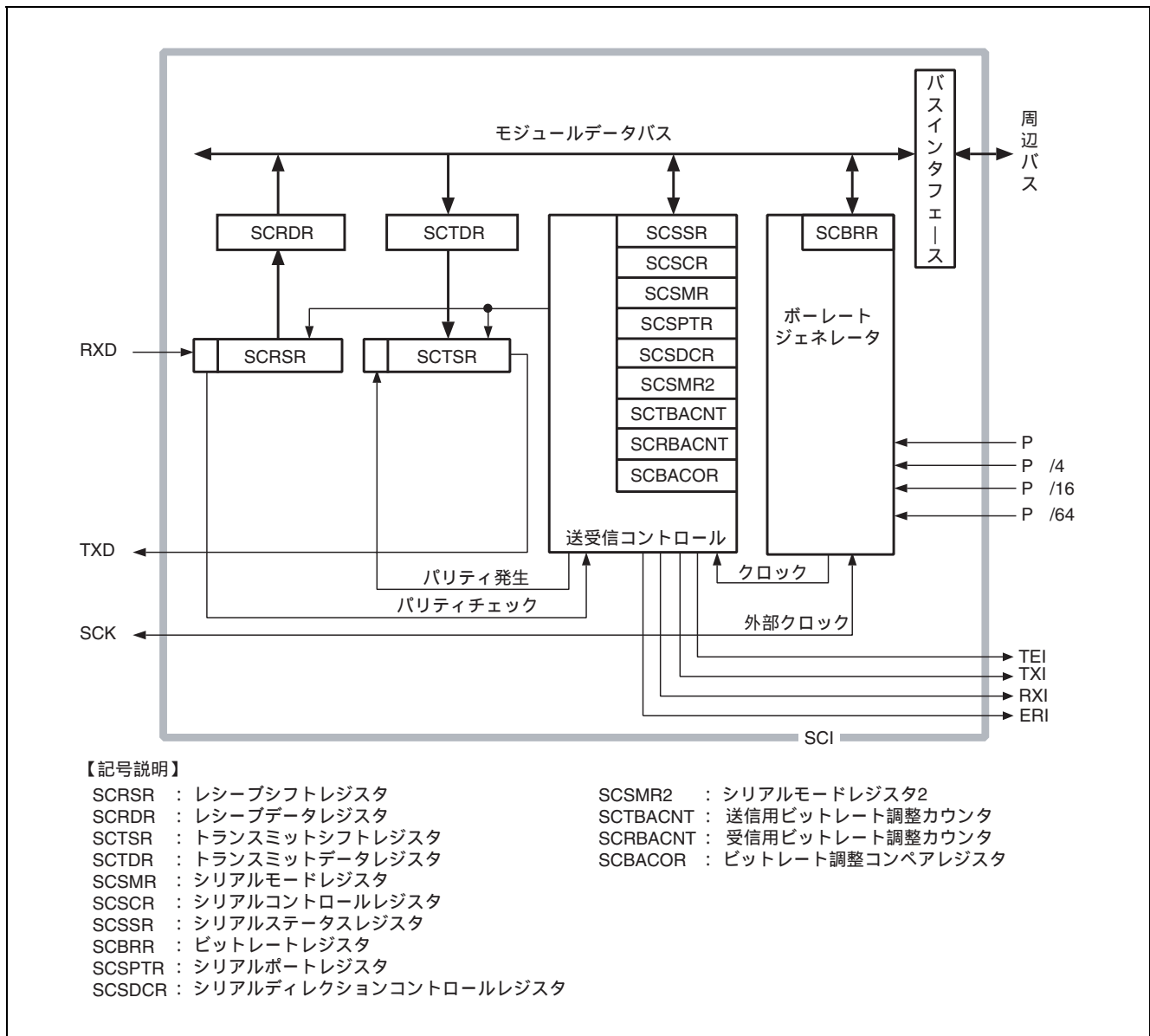


図4.6 SCIのブロック図

## 5. ハードウェアウェア説明

### 5.1 使用端子一覧

表 5.1 に使用端子と機能を示します。

表5.1 使用端子と機能

| 端子名      | 入出力 | 内容         |
|----------|-----|------------|
| PA1/TXD0 | 出力  | 調歩同期式送信データ |

## 6. ソフトウェア説明

### 6.1 動作概要

サンプルコードでは、DMAC と SCI の設定を行ってから、DMAC を有効にしています。SCI の初期設定時に、SCI のトランスミットデータレジスタ (SCTDR) は空の状態です。このため、DMAC を有効にすると、送信データエンプティによる DMA 転送要求によって、転送元メモリ領域から SCTDR への DMA 転送が開始されます。SCTDR に書き込まれたデータは、SCI に設定された通信フォーマットで TXD 端子から送信されます。

表 6.1 に DMAC の設定内容を、表 6.2 に SCI の設定内容を示します。また、図 6.1 に動作イメージを、図 6.2 に動作タイミングを示します。

表6.1 DMAC の設定内容

| 項目      | 内容                                 |
|---------|------------------------------------|
| 使用チャンネル | CH0                                |
| 転送データ長  | バイト                                |
| 転送回数    | 転送元となる文字列データのバイト数                  |
| アドレスモード | デュアルアドレスモード                        |
| 転送要求    | TXI0 (SCI チャンネル 0 の送信データエンプティ割り込み) |
| バスモード   | サイクルスチールモード (通常モード)                |
| 優先順位    | チャンネル優先順位固定モード                     |
| 割り込み    | 割り込み禁止                             |

表6.2 SCI の設定内容

| 項目      | 内容   |
|---------|--|
| 使用チャンネル | CH0  |
| 通信モード   | 調歩同期式  |
| データ長    | 8 ビット  |
| ストップビット | 1 ビット  |
| パリティ    | なし   |
| クロックソース | ポーレートジェネレータ (内部クロック)   |
| ポーレート   | 9600bps  |
| 割り込み    | TXI0 (送信データエンプティ割り込み) を有効<br>(ただし、DMA 転送要求の有効化のためであり、割り込みは未使用) |

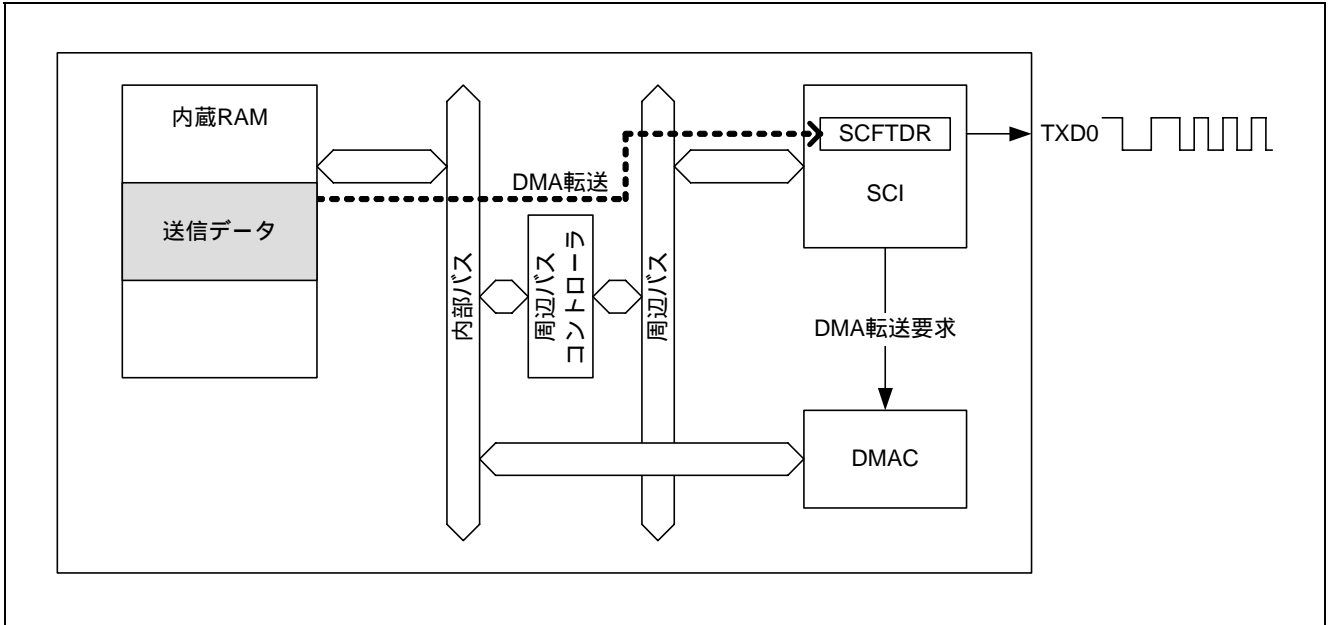


図6.1 動作イメージ

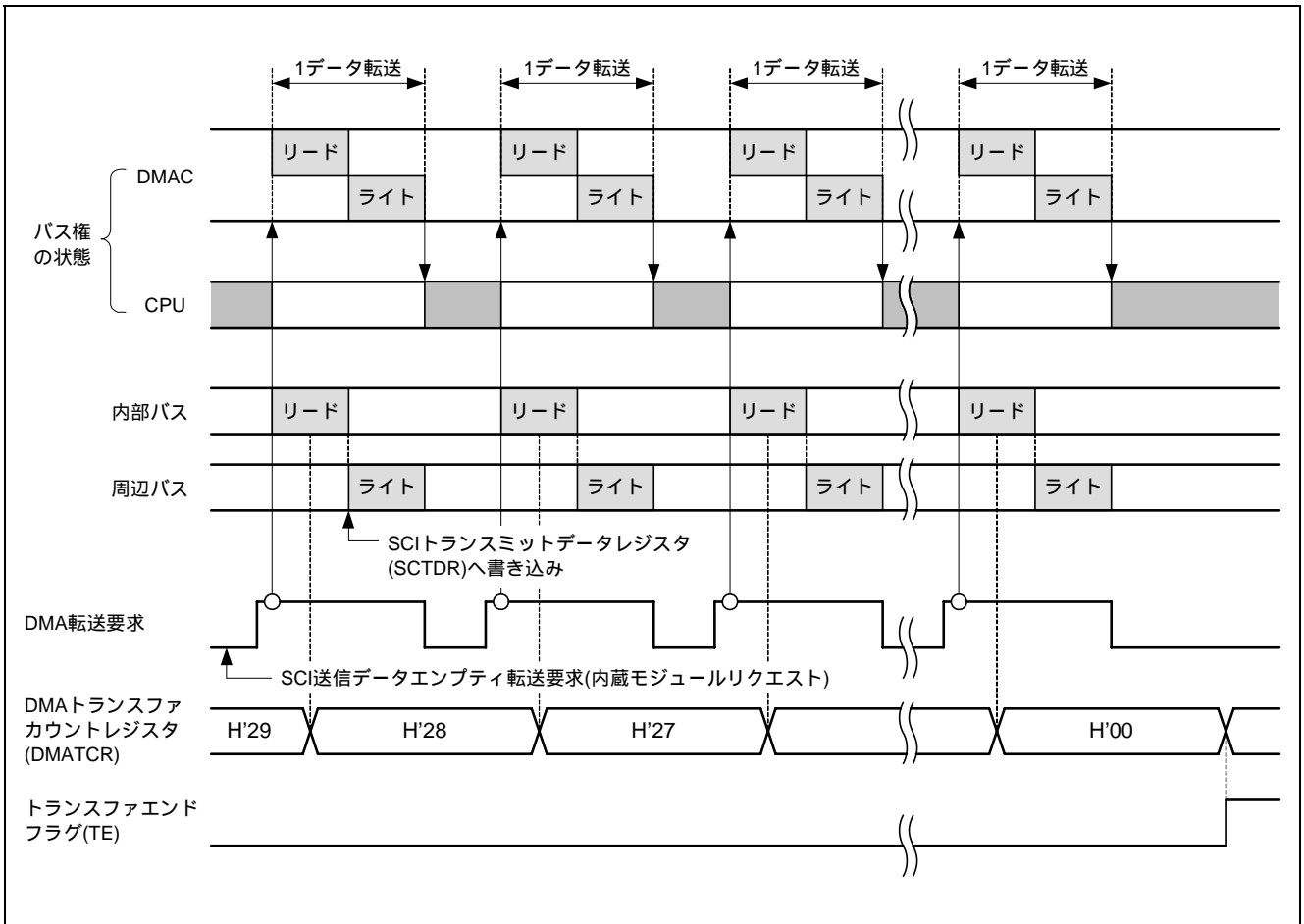


図6.2 動作タイミング

## 6.2 ファイル構成

表 6.3にサンプルコードで使用するファイルを示します。なお、統合開発環境で自動生成されるファイルは除きます。

表6.3 サンプルコードで使用するファイル

| ファイル名  | 概要       | 備考            |
|--------|----------|---------------|
| main.c | メインモジュール | 初期設定、DMA 転送処理 |

### 6.3 構造体/列挙体一覧

図 6.3 にサンプルコードで使用する構造体/列挙体を示します。

```
typedef enum
{
    DMA_SIZE_BYTE,          /* 8bit Transfer */
    DMA_SIZE_WORD,         /* 16bit Transfer */
    DMA_SIZE_LONG,         /* 32bit Transfer */
    DMA_SIZE_16B           /* 16Byte Transfer */
} dma_size_t;

typedef enum
{
    DMA_INT_DISABLE,
    DMA_INT_ENABLE
} dma_int_ctrl_t;

typedef struct
{
    dma_size_t      size;      /* size of transfer */
    dma_int_ctrl_t int_ctrl;  /* mode of interrupt */
} dma_mode_t;

typedef struct
{
    uint8_t  scbrr;          /* setting of SCBRR */
    uint8_t  scsmr;         /* setting of SCSMR */
} sci_baud_setting_t;

typedef enum
{
    BAUD_1200,             /* 1,200 bps */
    BAUD_2400,             /* 2,400 bps */
    BAUD_4800,             /* 4,800 bps */
    BAUD_9600,             /* 9,600 bps */
    BAUD_19200,            /* 19,200 bps */
    BAUD_31250,            /* 31,250 bps */
    BAUD_38400,            /* 38,400 bps */
    BAUD_57600,            /* 57,600 bps */
    BAUD_115200            /* 115,200 bps */
} sci_baud_t;
```

図6.3 サンプルコードで使用する構造体/列挙体

## 6.4 変数一覧

表 6.4にグローバル変数を、表 6.5にconst 型変数を示します。

表6.4 グローバル変数

| 型      | 変数名              | 内容                           | 使用関数 |
|--------|------------------|------------------------------|------|
| int8_t | g_send_strings[] | DMA 転送で SCI から送信する文字列を格納した配列 | main |

表6.5 const 型変数

| 型                  | 変数名                   | 内容                       | 使用関数        |
|--------------------|-----------------------|--------------------------|-------------|
| dma_mode_t         | g_dma_mode            | DMA 転送モードの設定値            | main        |
| sci_baud_setting_t | g_sci_baud_settings[] | 各通信速度に対応したボーレート設定用データの配列 | io_sci_init |

## 6.5 関数一覧

表 6.6に関数を示します。

表6.6 関数

| 関数名             | 概要            |
|-----------------|---------------|
| main            | メイン処理         |
| io_sci_init     | SCI の初期設定     |
| io_dma_init     | DMAC の初期設定    |
| io_dma_enable   | DMA 転送の有効化    |
| io_dma_poll_end | DMA 転送の完了待ち処理 |



## 6.6 関数仕様

サンプルコードの関数仕様を示します。

| main  |  |
|-------|--|
| 概要    | メイン処理  |
| ヘッダ   |  |
| 宣言    | void main(void)  |
| 説明    | DMAC と SCI の初期設定を行った後、DMA 転送を有効にし、転送完了を待ちます。最後に永久ループに入ります。 |
| 引数    | なし   |
| リターン値 | なし   |

| io_sci_init |  |
|-------------|--|
| 概要          | SCI の初期設定                              |
| ヘッダ         |  |
| 宣言          | void io_sci_init(sci_baud_t baudrate)  |
| 説明          | SCI のモジュールスタンバイを解除した後、SCI のレジスタ設定をします。 |
| 引数          | sci_baud_t baudrate : SCI の通信速度        |
| リターン値       | なし                                     |

| io_dma_init |   |
|-------------|---|
| 概要          | DMAC の初期設定  |
| ヘッダ         |   |
| 宣言          | void io_dma_init(uint32_t src_addr, uint32_t dst_addr,<br>uint32_t data_size, dma_mode_t mode)  |
| 説明          | DMAC のモジュールスタンバイを解除した後、DMAC のレジスタ設定をします。  |
| 引数          | uint32_t src_addr : 転送元メモリ領域のアドレス<br>uint32_t dst_addr : 転送先メモリ領域のアドレス<br>uint32_t data_size : 転送データサイズ<br>dma_mode_t mode : DMA 転送のモード設定 |
| リターン値       | なし  |

| io_dma_enable |                          |
|---------------|--------------------------|
| 概要            | DMA 転送の有効化               |
| ヘッダ           |                          |
| 宣言            | void io_dma_enable(void) |
| 説明            | DMA 転送を有効にします。           |
| 引数            | なし                       |
| リターン値         | なし                       |

---

io\_dma\_poll\_end

---

|       |                               |
|-------|-------------------------------|
| 概要    | DMA 転送の完了待ち処理                 |
| ヘッダ   |                               |
| 宣言    | void io_dma_poll_end(void)    |
| 説明    | DMA 転送の完了を待った後、DMA 転送を無効にします。 |
| 引数    | なし                            |
| リターン値 | なし                            |

## 6.7 フローチャート

### 6.7.1 メイン処理

図 6.4にメイン処理のフローチャートを示します。

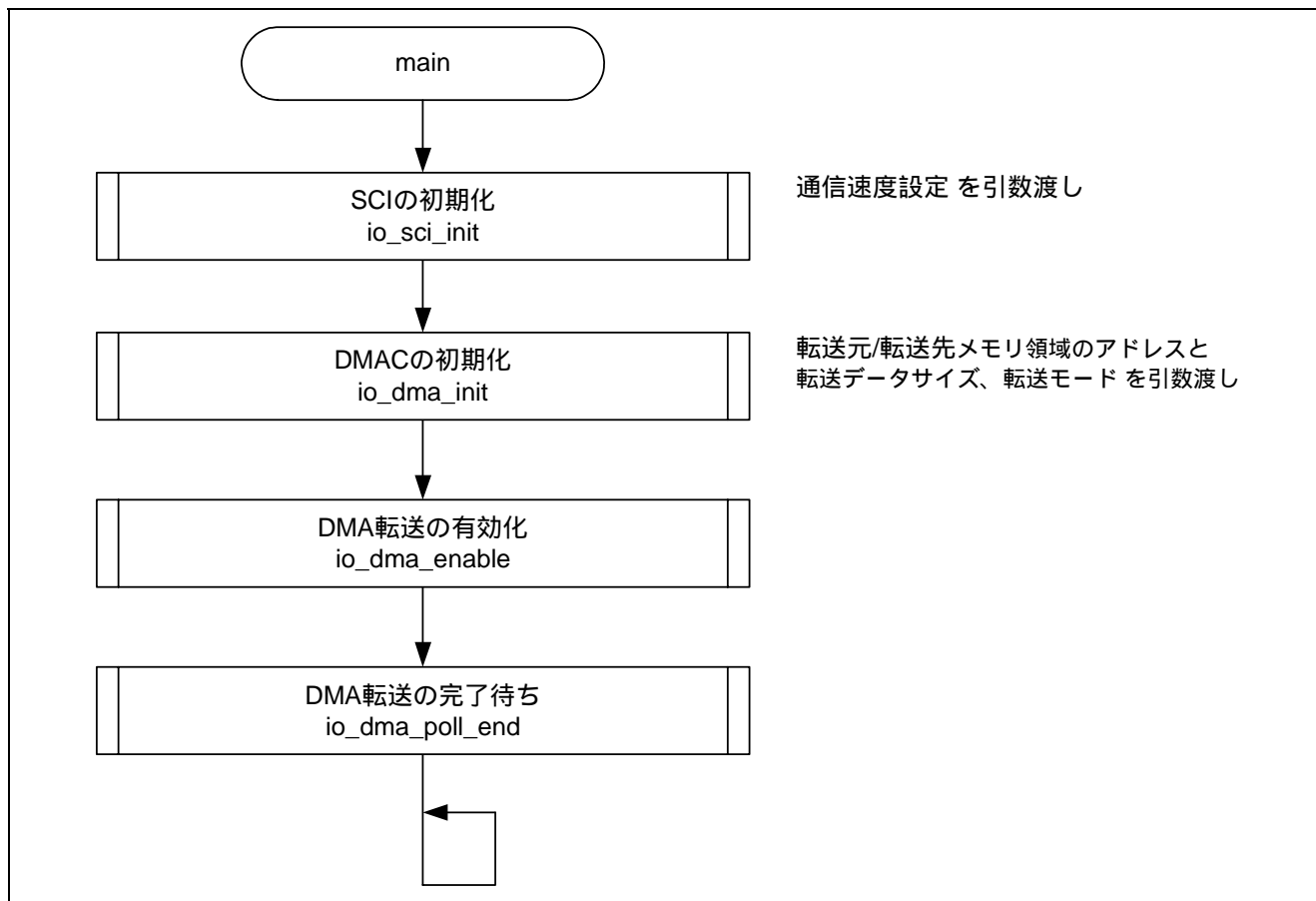


図6.4 メイン処理

## 6.7.2 SCI の初期設定

図 6.5にSCI の初期設定のフローチャートを示します。

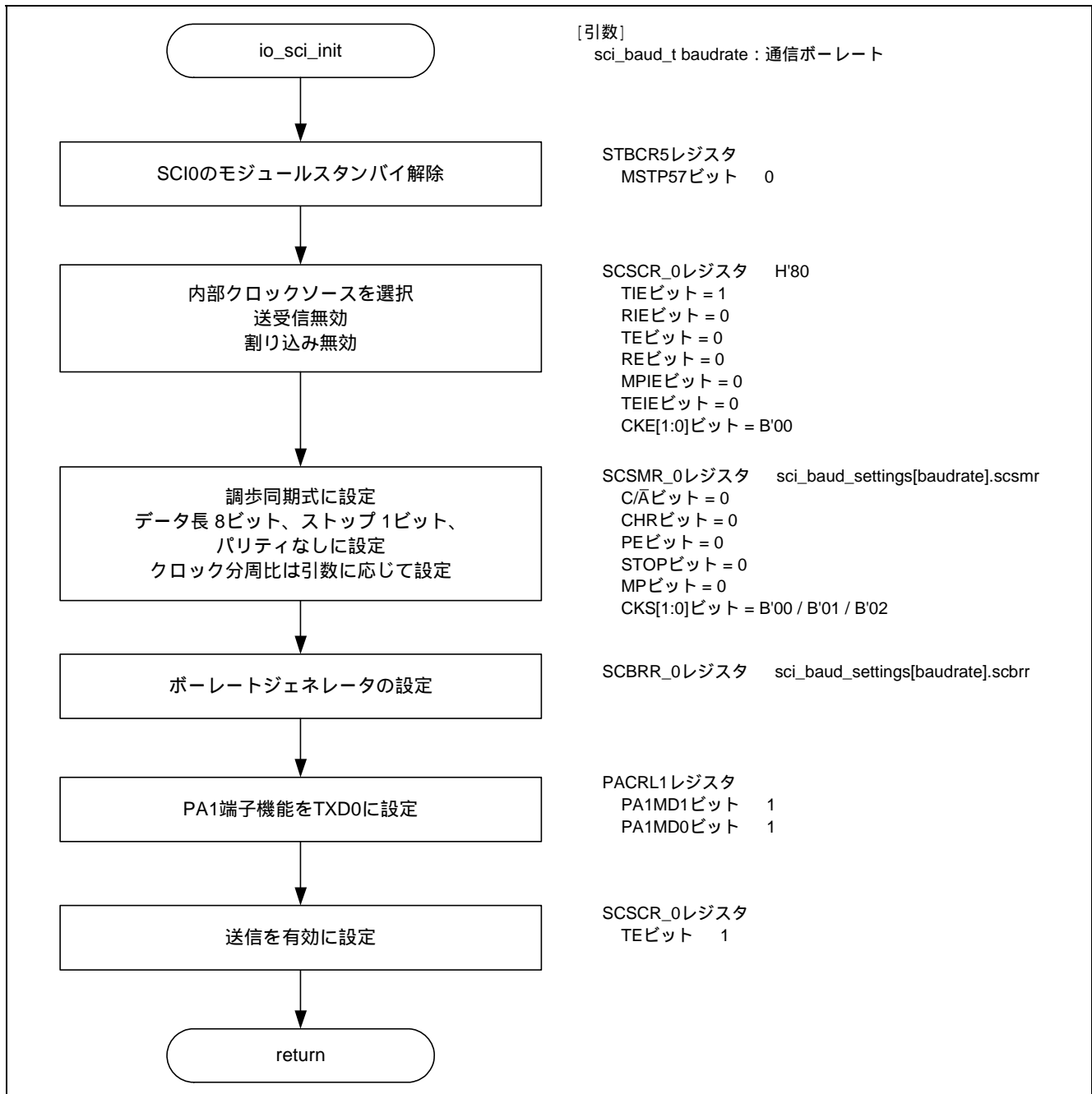


図6.5 SCI の初期設定

## 6.7.3 DMAC の初期設定

図 6.6にDMAC の初期設定のフローチャートを示します。

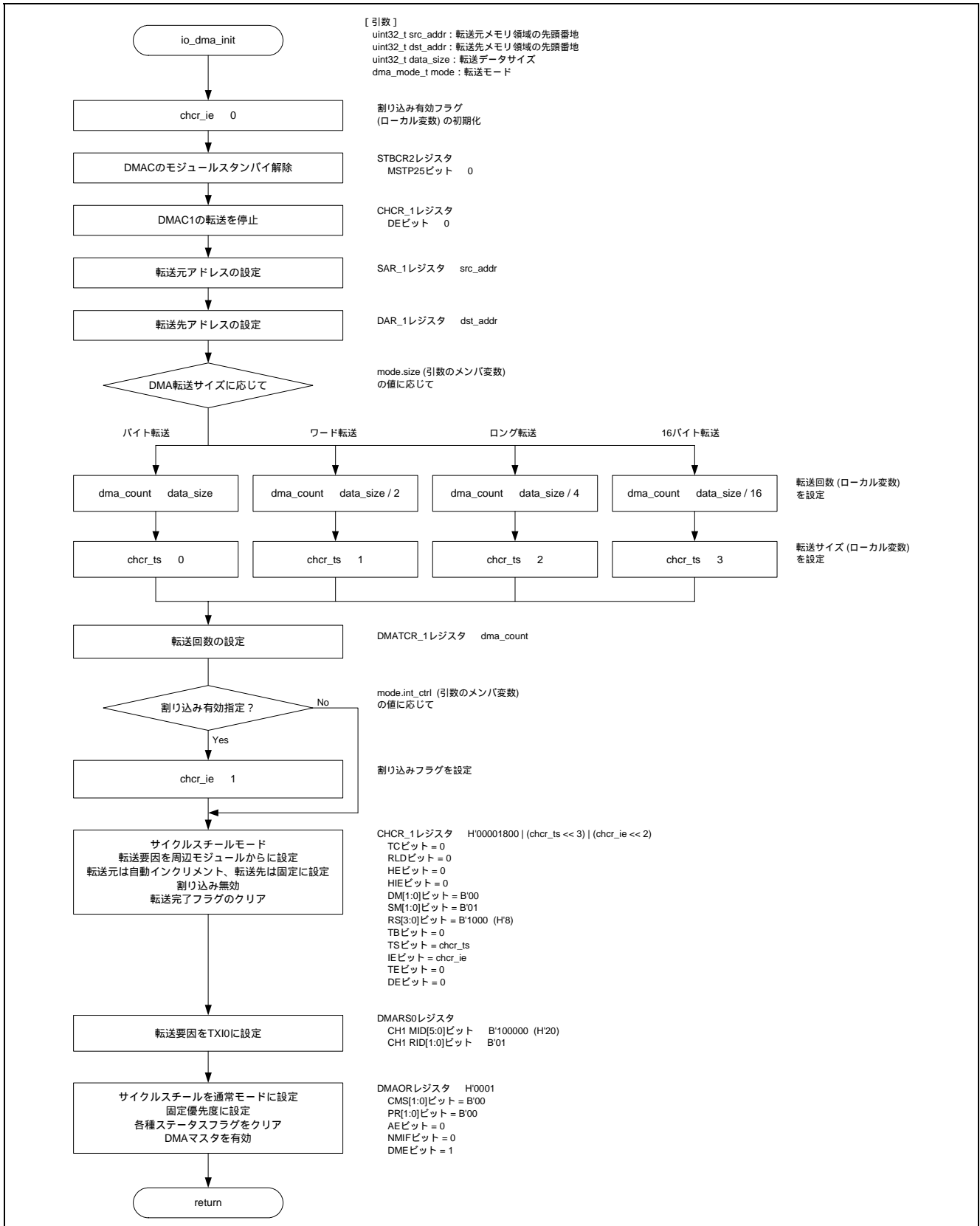


図6.6 DMAC の初期設定

## 6.7.4 DMA 転送の開始

図 6.7にDMA 転送の有効化のフローチャートを示します。

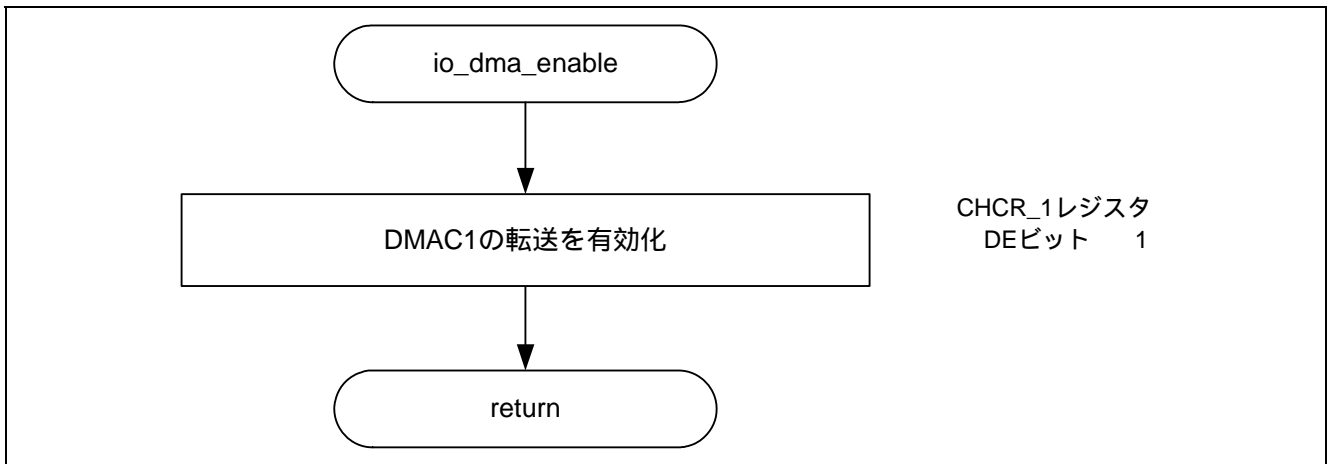


図6.7 DMA 転送の有効化

## 6.7.5 DMA 転送の完了待ち処理

図 6.8にDMA 転送の完了待ち処理のフローチャートを示します。

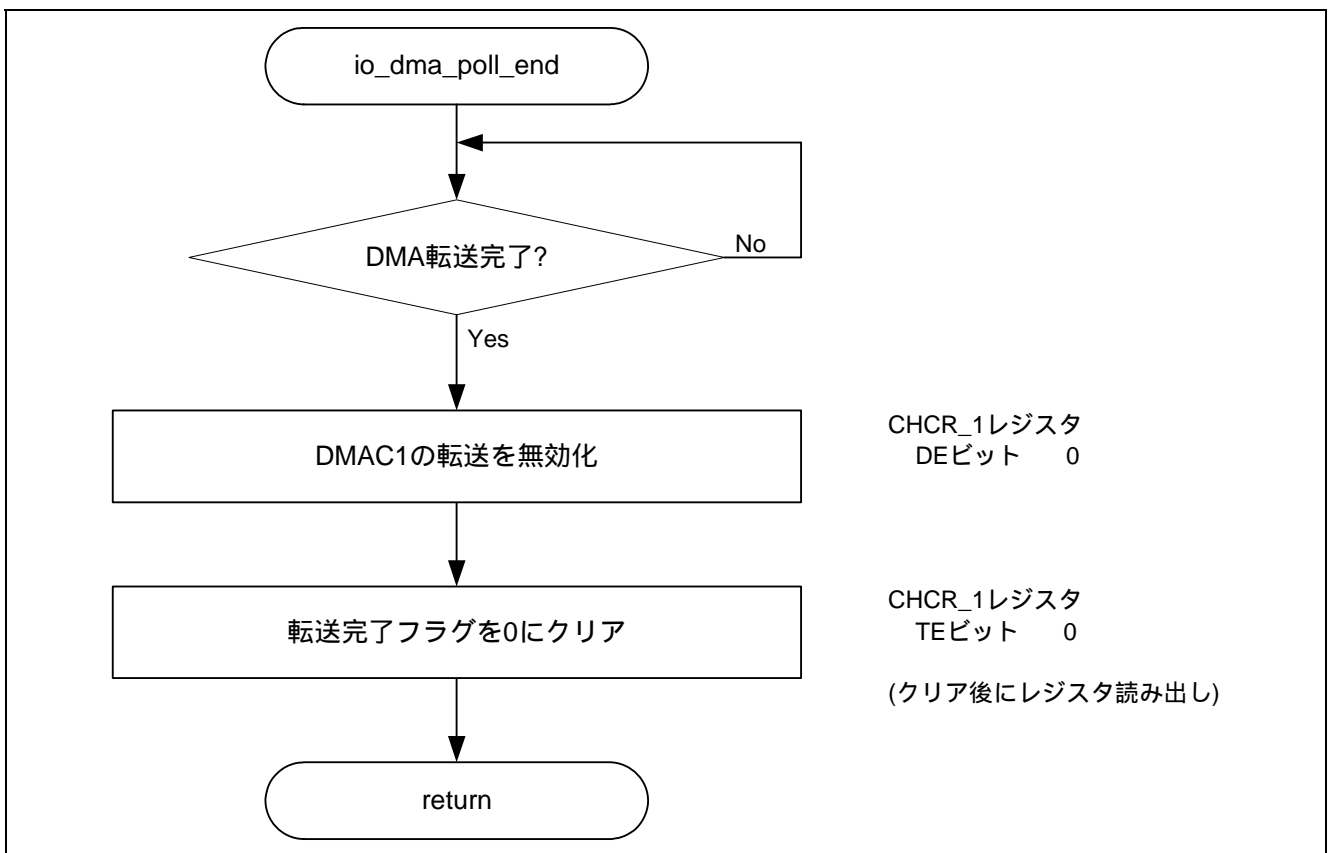


図6.8 DMA 転送の完了待ち処理

## 7. 参考ドキュメント

- ハードウェアマニュアル  
SH7231グループ ユーザーズマニュアル ハードウェア編 Rev.2.00 ( R01UH0073JJ )  
( 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。 )
- ソフトウェアマニュアル  
SH-2A、SH2A-FPU ユーザーズマニュアル ソフトウェア編 Rev.4.00 ( R01US0031JJ )  
( 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/contact/>

## 改訂記録

| Rev. | 発行日        | 改訂内容 |      |
|------|------------|------|------|
|      |            | ページ  | ポイント |
| 1.00 | 2012.12.03 | —    | 新規作成 |
|      |            |      |      |
|      |            |      |      |
|      |            |      |      |



## 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・システムの設計において、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て、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。
2. 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。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因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に対する侵害に関し、当社は、何らの責任を負うものではありません。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。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等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。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。  
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、産業用ロボット等  
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、各種安全装置等  
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図し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損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の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。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、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、社会的損害等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、お客様の機器・システムとしての出荷保証を行ってください。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、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の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て、当社は、一切その責任を負いません。
9. 本資料に記載されている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・システムに使用することはできません。また、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に使用しないでください。当社製品または技術を輸出する場合は、「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。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にご負担して頂きますのでご了承ください。
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禁じます。

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/>