

SH7216 グループ

R01AN0039JJ0100

Rev.1.00

2010.07.26

ルネサスシリアルペリフェラルインタフェース マルチメディアカードのアクセス例

要旨

本アプリケーションノートでは、ルネサスシリアルペリフェラルインタフェースを使用したマルチメディアカード（MMC）へのアクセス例について説明しています。

動作確認デバイス

SH7216

目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. 参考プログラム.....	15
4. 参考ドキュメント.....	24

1. はじめに

1.1 仕様

ルネサスシリアルペリフェラルインタフェースを使用して、MMC にアクセスします。

1.2 使用機能

- ルネサスシリアルペリフェラルインタフェース
- ピンファンクションコントローラ

1.3 適用条件

マイコン	SH7216
動作周波数	内部クロック : 200 MHz バスクロック : 50 MHz 周辺クロック : 50 MHz AD クロック : 50 MHz
統合開発環境	ルネサス エレクトロニクス製 High-performance Embedded Workshop Ver.4.07.00
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 C/C++コンパイラパッケージ Ver.9.03 Release 00
コンパイルオプション	High-performance Embedded Workshop でのデフォルト設定 (-cpu=sh2afpu -fpu=single -debug -gbr=auto -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1)
MMC	Version 4.2 対応品 【注】 Version 4.3 以降は SPI モードに対応していません。

1.4 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。合わせて参照してください。

- SH7216 グループ 初期設定例

1.5 "L"アクティブ端子（信号）の表記について

端子名（信号名）末尾の#は“L”アクティブ端子（信号）であることを示します。

2. 応用例の説明

本応用例では、ルネサスシリアルペリフェラルインタフェース（RSPI）を使用して、MMC にアクセスします。

ここでは MMC のプロトコル概要と、ルネサスシリアルペリフェラルインタフェース（RSPI）について説明します。MMC の詳細については MMC の規格書を参照してください。

2.1 MMC のプロトコル概要

2.1.1 SPI モード

MMC の動作モードには、本来の動作モードである MMC モードの他に、シリアルペリフェラルインタフェース（SPI）で動作する SPI モードがあります。本応用例ではこの SPI モードを使用します。以降の説明では SPI モードの使用を前提としているため、MMC モードについての説明は省略します。

2.1.2 コマンド

MMC へのアクセスには、コマンドを使用します。

図 1 にコマンドフォーマットを示します。Start、Host、End ビットはそれぞれ 0、1、1 に固定され、コマンドコードにはコマンド名称に含まれる数値が設定されます。コマンドパラメータにはコマンドに応じて引数を設定します。CRC7 は、コマンドコードとコマンドパラメータを含む 5 バイトに対して演算された CRC が設定されます。

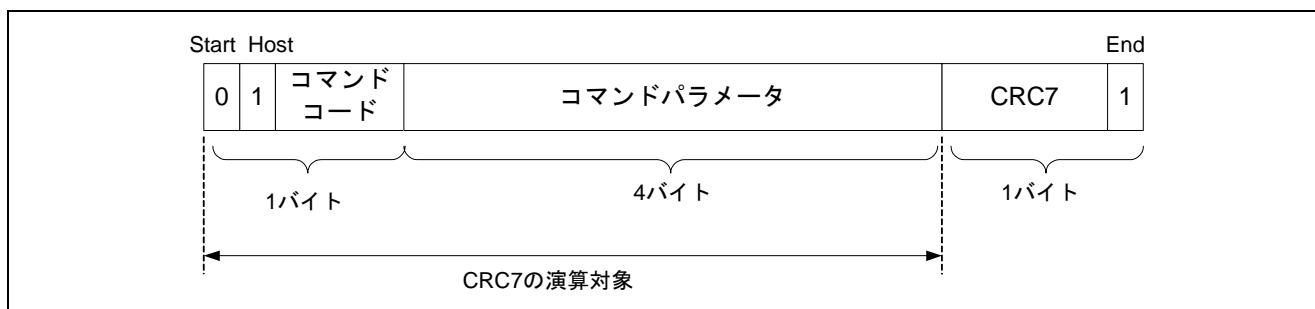


図1 コマンドフォーマット

表 1 に主要なコマンドを示します。

表1 主要なコマンド

名称	コマンドコード	機能	名称	コマンドコード	機能
CMD0	H'00	MMC リセットを要求	CMD17	H'11	シングルブロックリード
CMD1	H'01	MMC 初期化プロセス実行	CMD18	H'12	マルチブロックリード
CMD9	H'09	CSD レジスタの読み出し	CMD24	H'18	シングルブロックライト
CMD10	H'0A	CID レジスタの読み出し	CMD25	H'19	マルチブロックライト
CMD12	H'0C	マルチブロックリードの停止	CMD58	H'3A	OCR レジスタの読み出し
CMD13	H'0D	ステータスレジスタの読み出し	CMD59	H'3B	CRC の ON/OFF 設定

2.1.3 レスポンス

コマンドを受信した MMC はレスポンスを返します。レスポンスのフォーマットはコマンドに応じて異なりますが、ここでは主要なレスポンスとして R1 レスポンスを説明します。

図 2 に R1 レスポンスのフォーマットを示します。

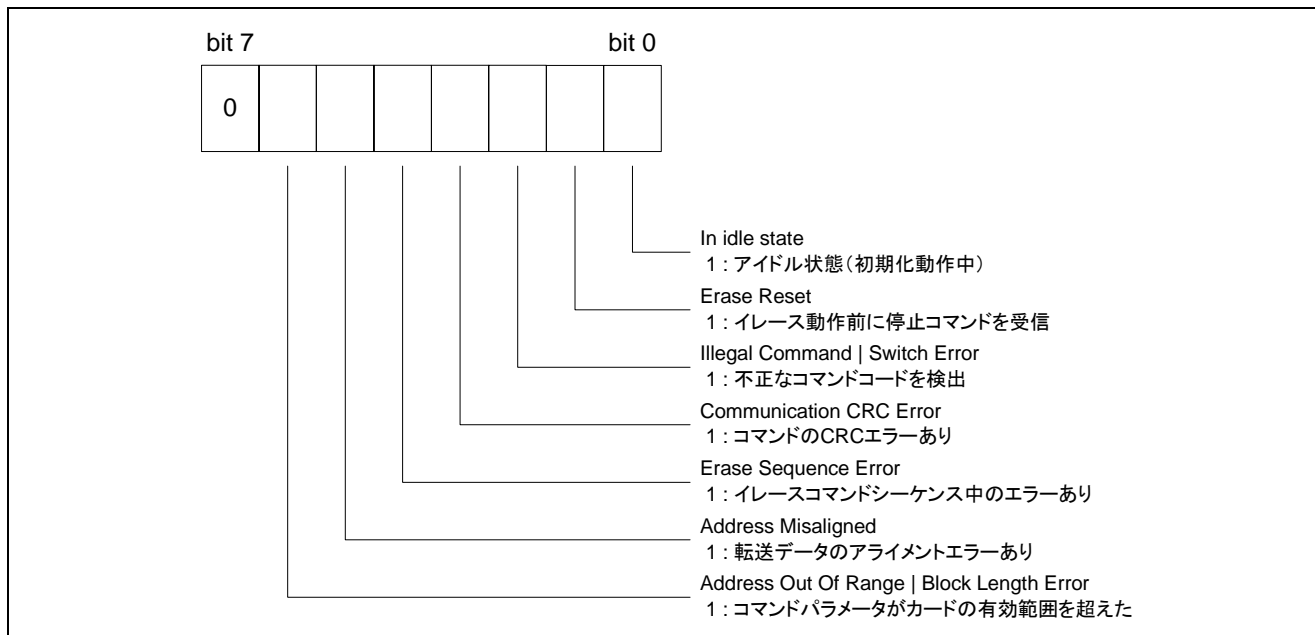


図2 R1 レスポンス

2.1.4 データトークン

データをリードまたはライトするコマンドの場合、レスポンスの受信後に、データトークンを使用してデータを転送します。

図3にデータトークンフォーマットを示します。データの先頭にはスタートデータブロックトークンが挿入されます。これはアイドル時に受信される無効な値 (H'FF) と有効データの H'FF を識別するためです。なお、データの終わりに CRC16 が付加されますが、スタートデータブロックトークンは演算対象に入りません。表2にスタートデータブロックトークンの種類を示します。

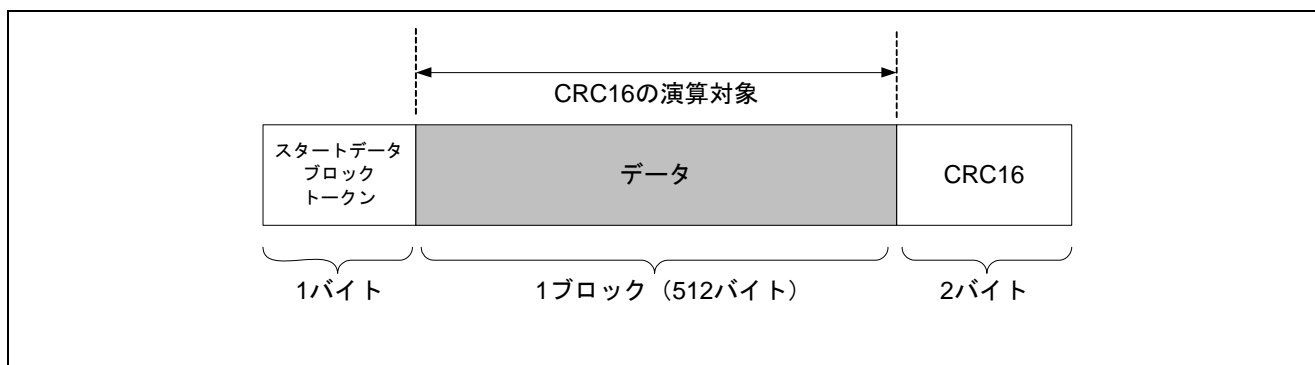


図3 データトークンフォーマット

表2 スタートデータブロックトークンの種類

データトークンの種類	値
シングル/マルチブロックリード	H'FE (B'1111 1110)
シングルブロックライト	H'FE (B'1111 1110)
マルチブロックライト (開始)	H'FC (B'1111 1100)
マルチブロックライト (停止)	H'FD (B'1111 1101)

図4にデータエラートークンのフォーマットを示します。データリード時に MMC がデータを転送できない場合は、データトークンの代わりに、MMC がこのデータエラートークンを返します。

なお、データライト時は、正常にライトできたかどうかを示すデータレスポンスを返します。

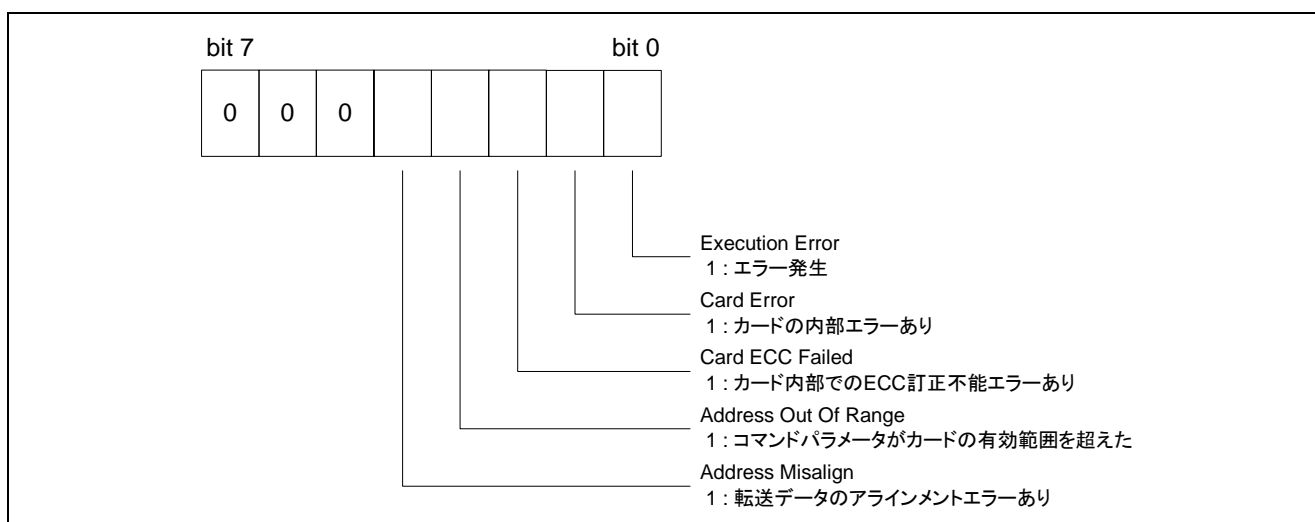


図4 データエラートークンのフォーマット

2.1.5 動作シーケンス

ここではMMCアクセスの動作シーケンスを説明します。

図5にデータがない場合の動作シーケンスを示します。コマンド発行後にレスポンスを受信します。MMCのアイドル時に受信しようとする H'FF を受信します。その場合は受信データを破棄してください。また、ビジー状態が発生する場合は受信データがビジー状態 (H'00) からアイドル状態 (H'FF) に変わるまで待ちます。

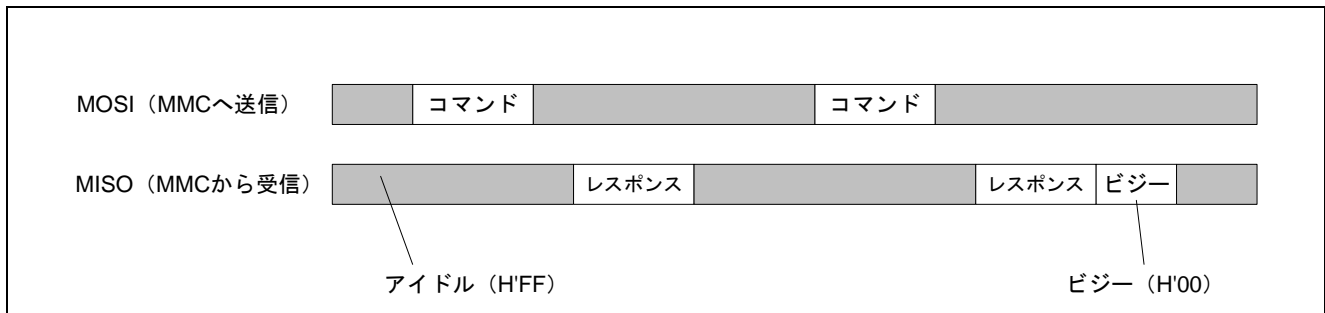


図5 データがない場合の動作シーケンス

図6にマルチブロックリード時のデータリードシーケンスを示します。マルチブロックリードコマンド発行後にレスポンスを受信し、続いてデータトークンを受信します。停止する場合は、ストップコマンドを発行します。

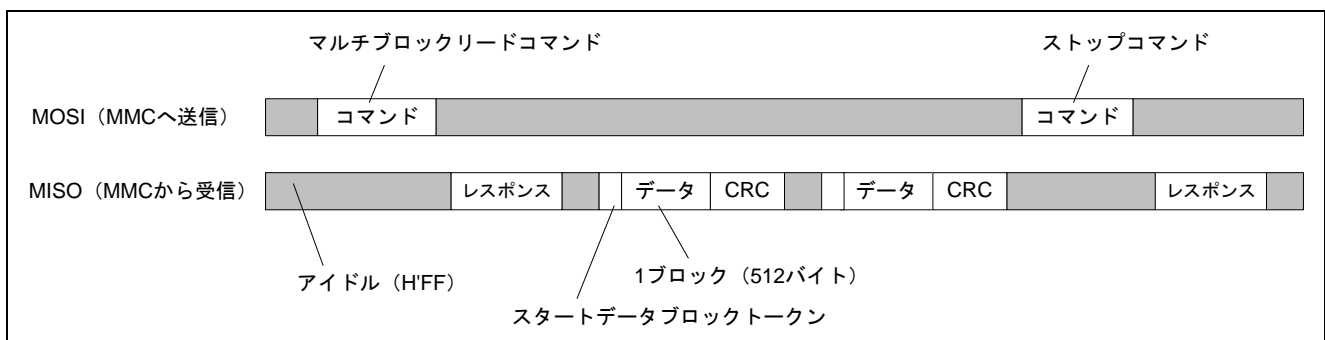


図6 データリードシーケンス

図7にマルチブロックライト時のデータライトシーケンスを示します。マルチブロックライトコマンド発行後にレスポンスを受信し、続いてデータトークンを送信します。データトークンの送信後はデータレスポンスを受信します。停止する場合は、マルチブロックライト停止のデータトークンを発行します。

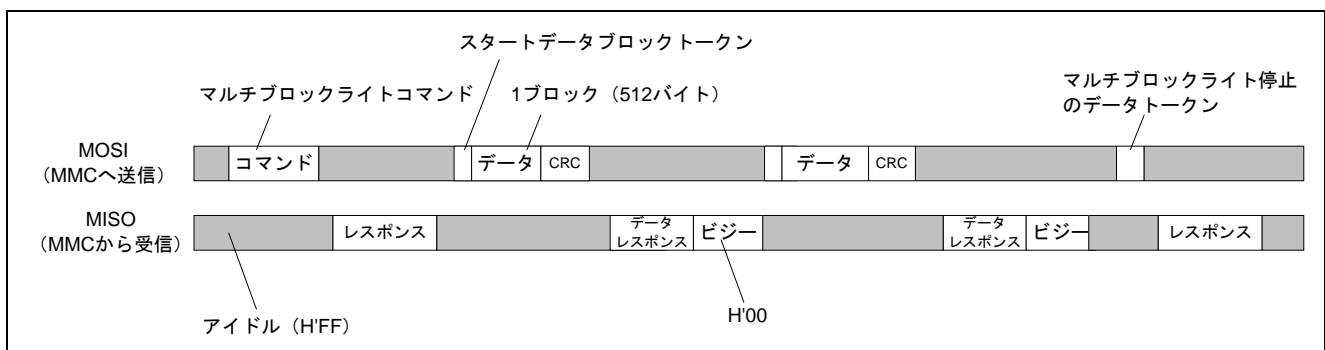


図7 データライトシーケンス

2.1.6 CRC

MMC は CRC7 と CRC16 を使用して誤り検出を行います。図 8 と図 9 に、CRC7 および CRC16 の概要を示します。

SPI モードの初期設定では CRC が無効ですが、CMD59 で有効/無効を設定できます。なお、CRC が無効な場合でも CRC の送受信は省略できません。

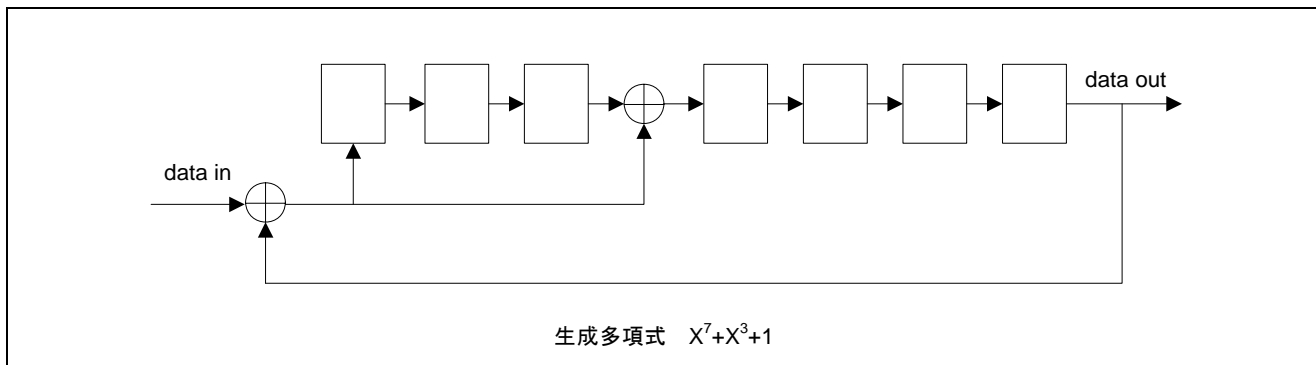


図8 CRC7 の概要

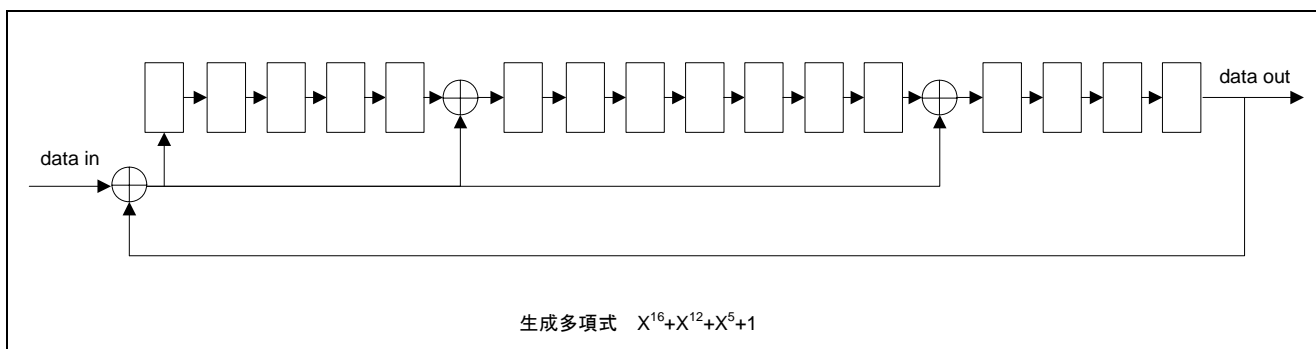


図9 CRC16 の概要

2.1.7 MMC の初期化手順

MMC は、パワーオン後に初期化シーケンスとして遅延サイクルの挿入が必要です。その後、CS#信号をアサートして CMD0 を発行すると SPI モードに遷移します。CMD1 で初期化を実行し、その初期化が完了するとデータ転送が可能になります。

図 10に SPI モード使用時の初期化手順を示します。

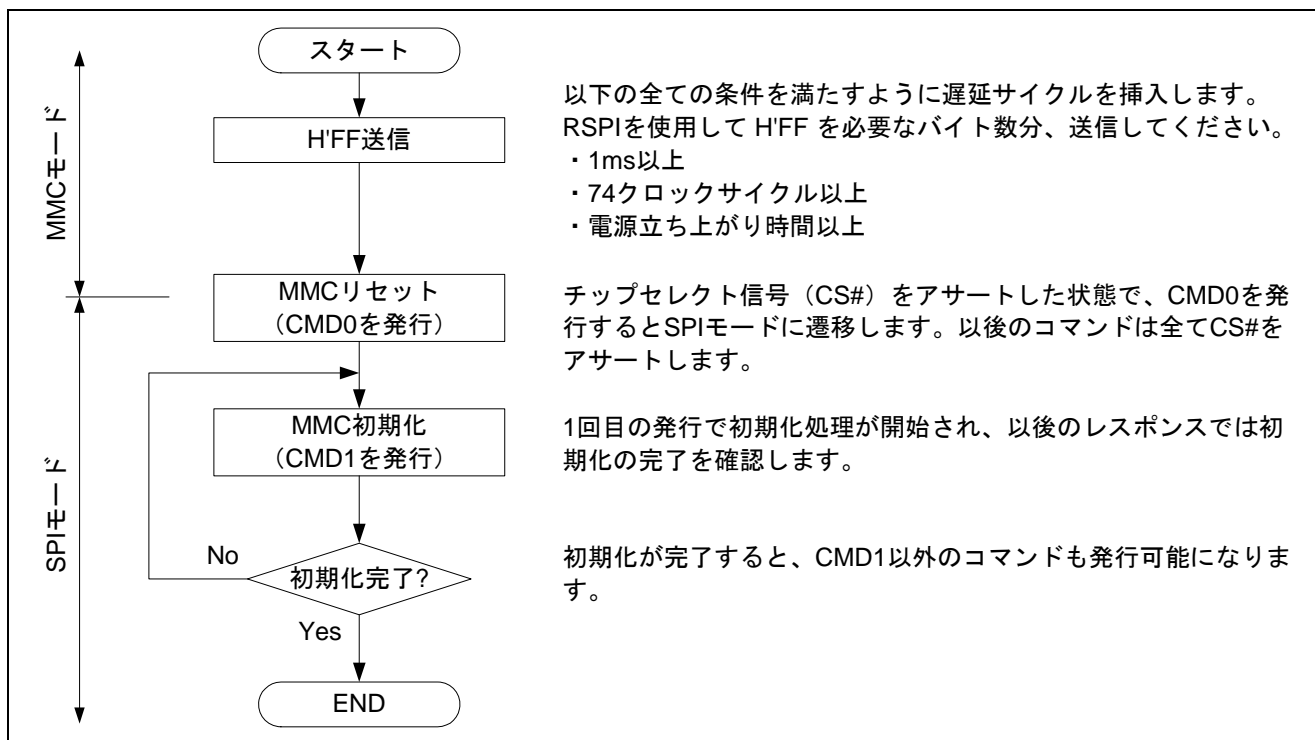


図10 SPIモード使用時の初期化手順

2.2 ルネサスシリアルペリフェラルインタフェース (RSPI)

ここでは、SH7216 内蔵のルネサスシリアルペリフェラルインタフェース (RSPI) の使用方法を説明します。

2.2.1 動作概要

SH7216 は、1 チャンネルのルネサスシリアルペリフェラルインタフェース (RSPI) を備えています。MOSI (Master Out Slave In) 端子および MISO (Master In Slave Out) 端子、SSL (Slave Select) 端子、RSPCK (SPI Clock) 端子を使用して、SPI 動作 (4 線式) / クロック同期式動作 (3 線式) でシリアル通信が可能です。RSPI は、マスタ/スレーブの選択、シリアル転送クロックの極性と位相の変更 (SPI モード変更)、転送ビット長の変更 (8、9、10、11、12、13、14、15、16、20、24、32 ビット) が可能なため、多様な SPI 互換デバイスを接続することができます。

図 11 に RSPI のブロック図を示します。

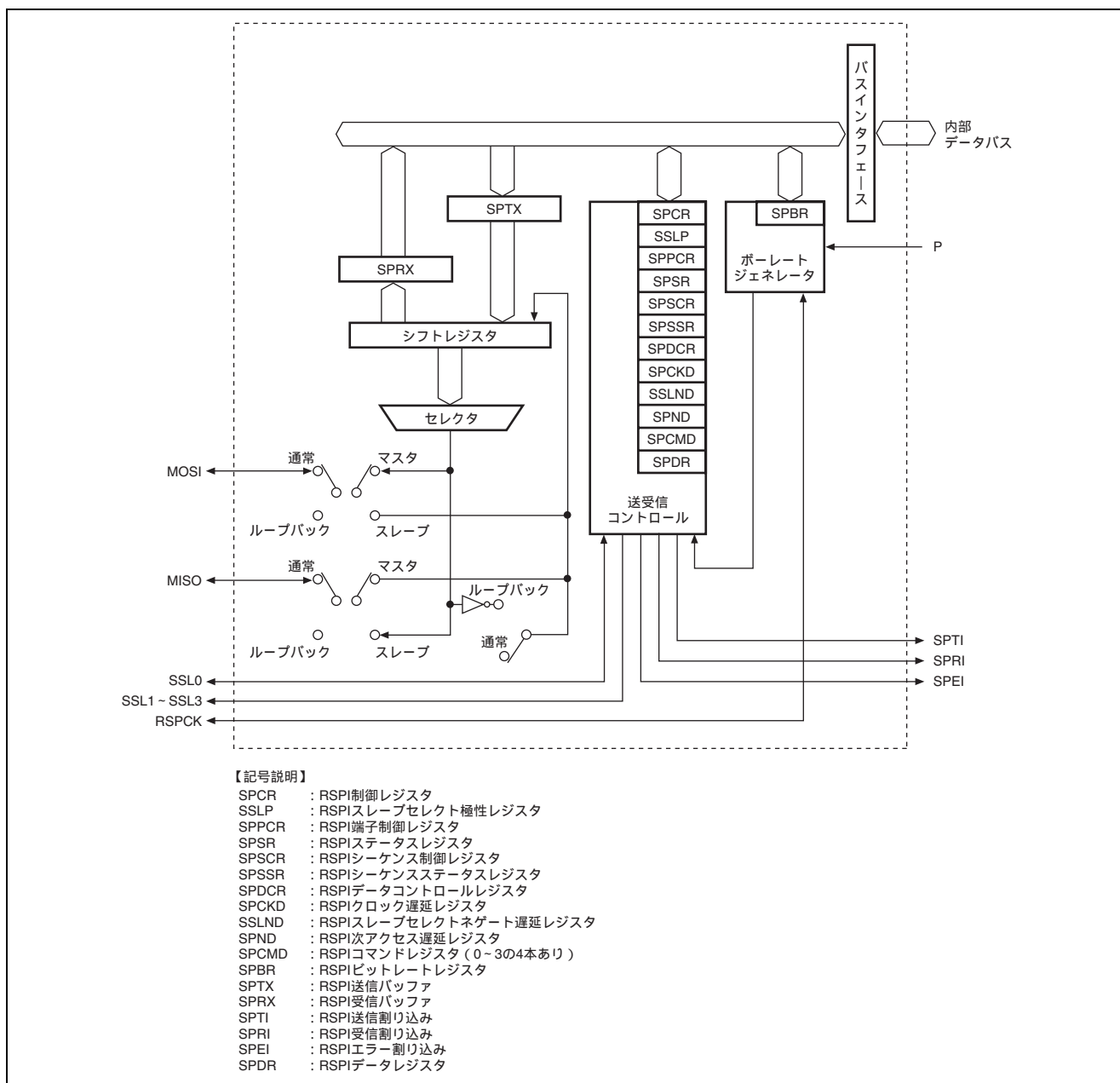


図11 RSPIのブロック図

2.2.2 端子接続例

表 3に MMC の端子一覧を示します。MMC モードと SPI モードは端子構成が異なり、パワーオン時は MMC モードになっています。ただし、SPI モードに遷移するまでは SPI モードと共通の端子しか使用しないため、SPI モードの端子接続で通信することが可能です。

表3 MMC の端子一覧

Pin No.	SPI モード		MMC モード	
	名称	機能	名称	機能
1	CS#	チップセレクト	DAT3	データ
2	DI	データイン	CMD	コマンド/レスポンス
3	VSS1	グラウンド	VSS1	グラウンド
4	VDD	電源	VDD	電源
5	SCLK	クロック	CLK	クロック
6	VSS2	グラウンド	VSS2	グラウンド
7	DO	データアウト	DAT0	データ
8	未使用		DAT1	データ
9	未使用		DAT2	データ
10	未使用		DAT4	データ
11	未使用		DAT5	データ
12	未使用		DAT6	データ
13	未使用		DAT7	データ

図 12 に SPI モードを使用した MMC 接続回路例を示します。マイコンの端子状態がハイインピーダンスの場合でも MMC が誤動作しないように、SSL3 端子および MOSI 端子、MISO 端子には外付けプルアップ抵抗を接続しています。また、コネクタの CD 端子は MMC の挿抜に応じて状態が変化する端子です。この信号を利用して SH7216 は MMC を検出しますが、図 12 に示す接続例は活線挿抜には対応していないため注意してください。

SH7216 の端子機能は表 4 のピンファンクションコントローラ (PFC) 設定例を参考に設定してください。

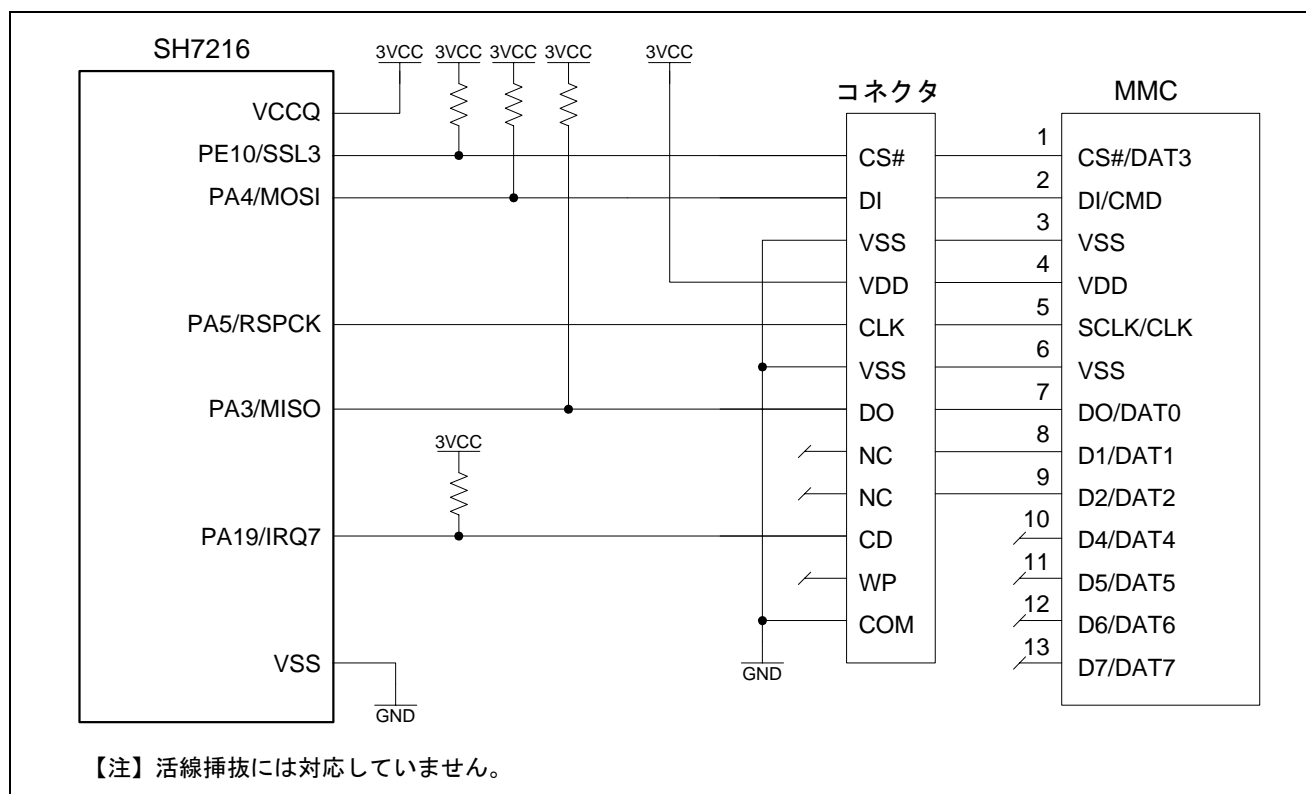


図12 SPI モードを使用した MMC 接続回路例

表4 ピンファンクションコントローラ (PFC) 設定例

レジスタ名	設定値	機能
ポート A コントロール レジスタ H1 (PACRH1)	H'0000	PA19 端子を汎用入出力ポートに設定 (IRQ7 割り込みを使用した接続検出も可能ですが、ここでは入力ポートを使用して接続を検出します。)
ポート A・IO レジスタ H (PAIORH)	H'0000	PA19 端子を入力に設定
ポート A コントロール レジスタ L2 (PACRL2)	H'0055	PA5 端子を RSPCK 入出力 (RSPI) に設定 PA4 端子を MOSI 入出力 (RSPI) に設定
ポート A コントロール レジスタ L1 (PACRL1)	H'5000	PA3 端子を MISO 入出力 (RSPI) に設定
ポート E コントロール レジスタ L3 (PECRL3)	H'0000	PE10 端子を汎用入出力ポートに設定 (ここでは RSPI の SSL 端子機能は使用せずに、出力ポートを制御してチップセレクト信号を生成します。)
ポート E・IO レジスタ L (PEIORL)	H'0400	PE10 端子を出力に設定

2.2.3 RSPI の初期設定手順

図 13および図 14に本参考プログラムにおける RSPI 初期設定フローを示します。この設定によりマスターモードでの SPI 動作が可能となります。

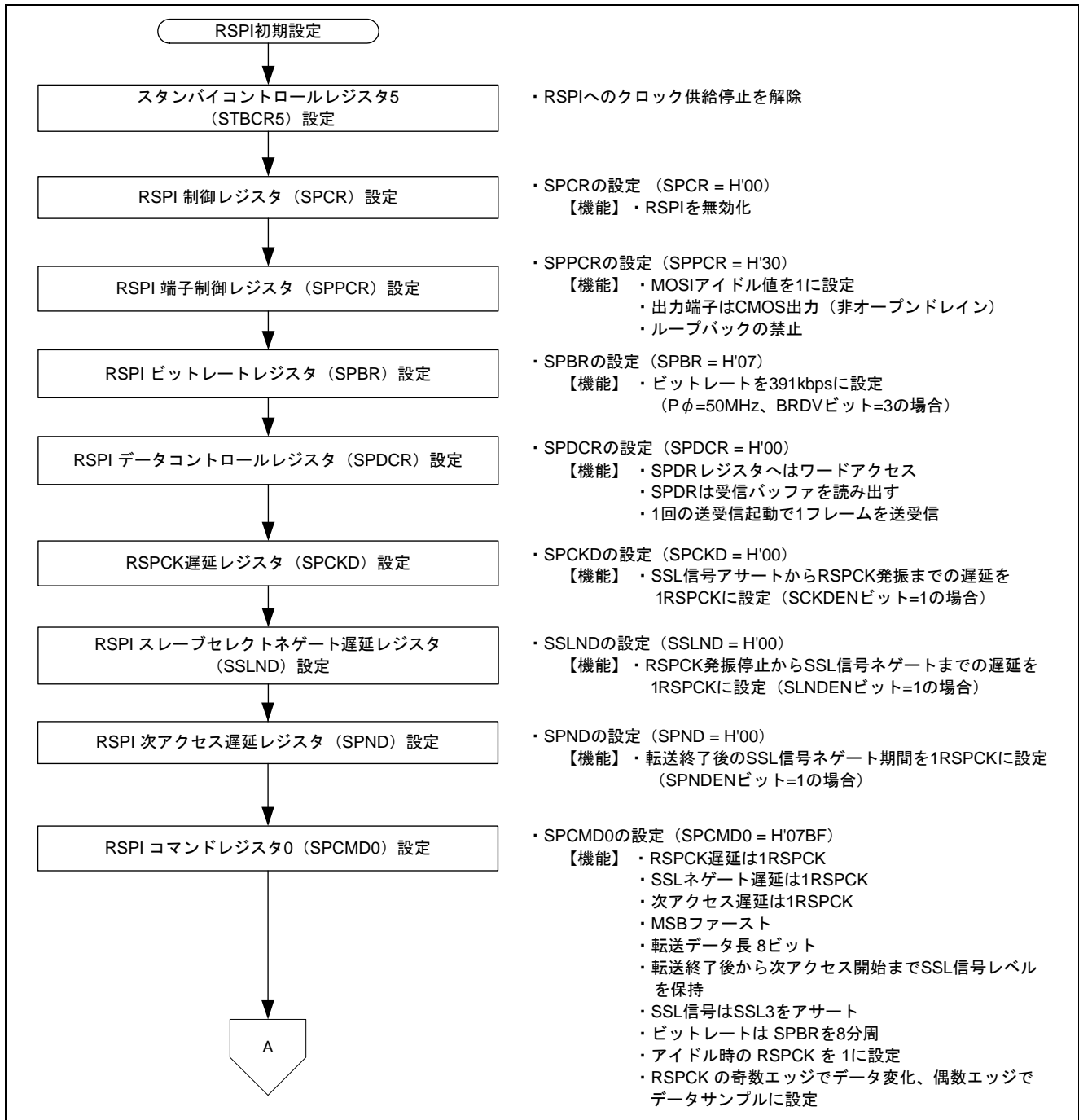


図13 参考プログラムの RSPI 初期設定フロー (1)

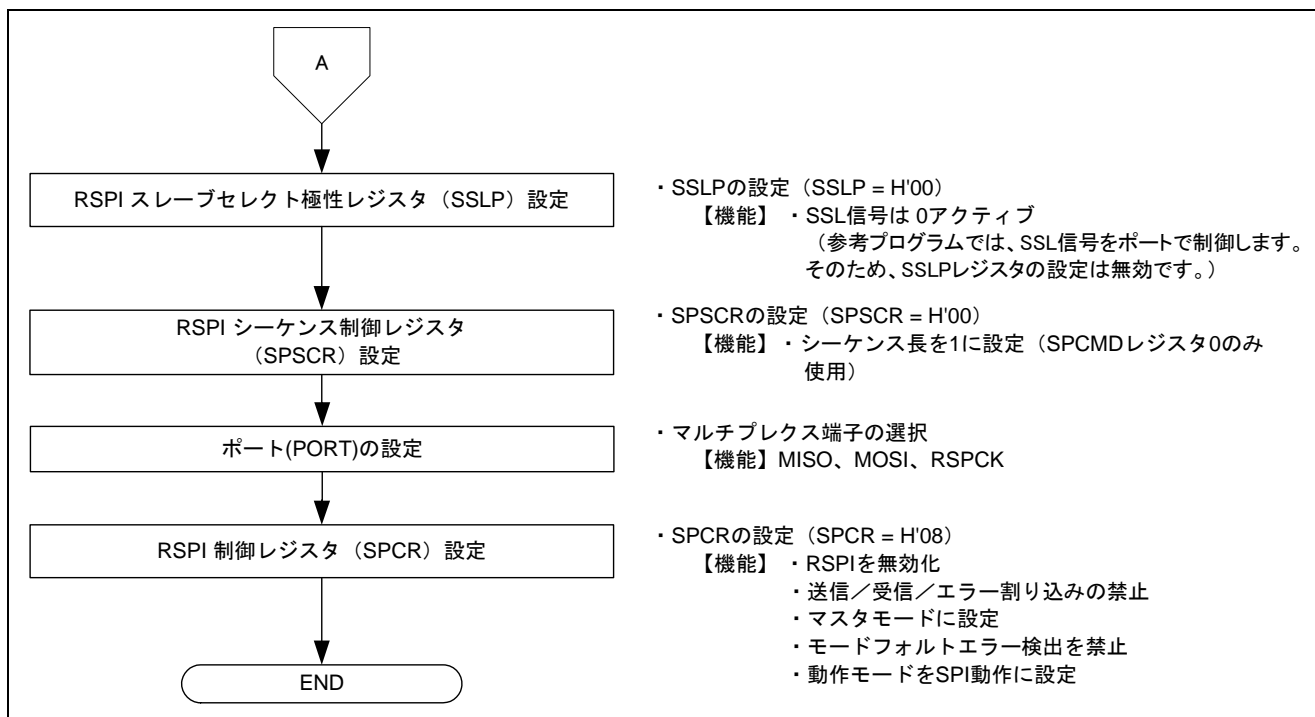


図14 参考プログラムのRSPI初期設定フロー (2)

2.2.4 RSPI のデータ転送手順

図 15に参考プログラムのデータ転送手順フローを示します。本手順により、全二重転送を行うことができます。

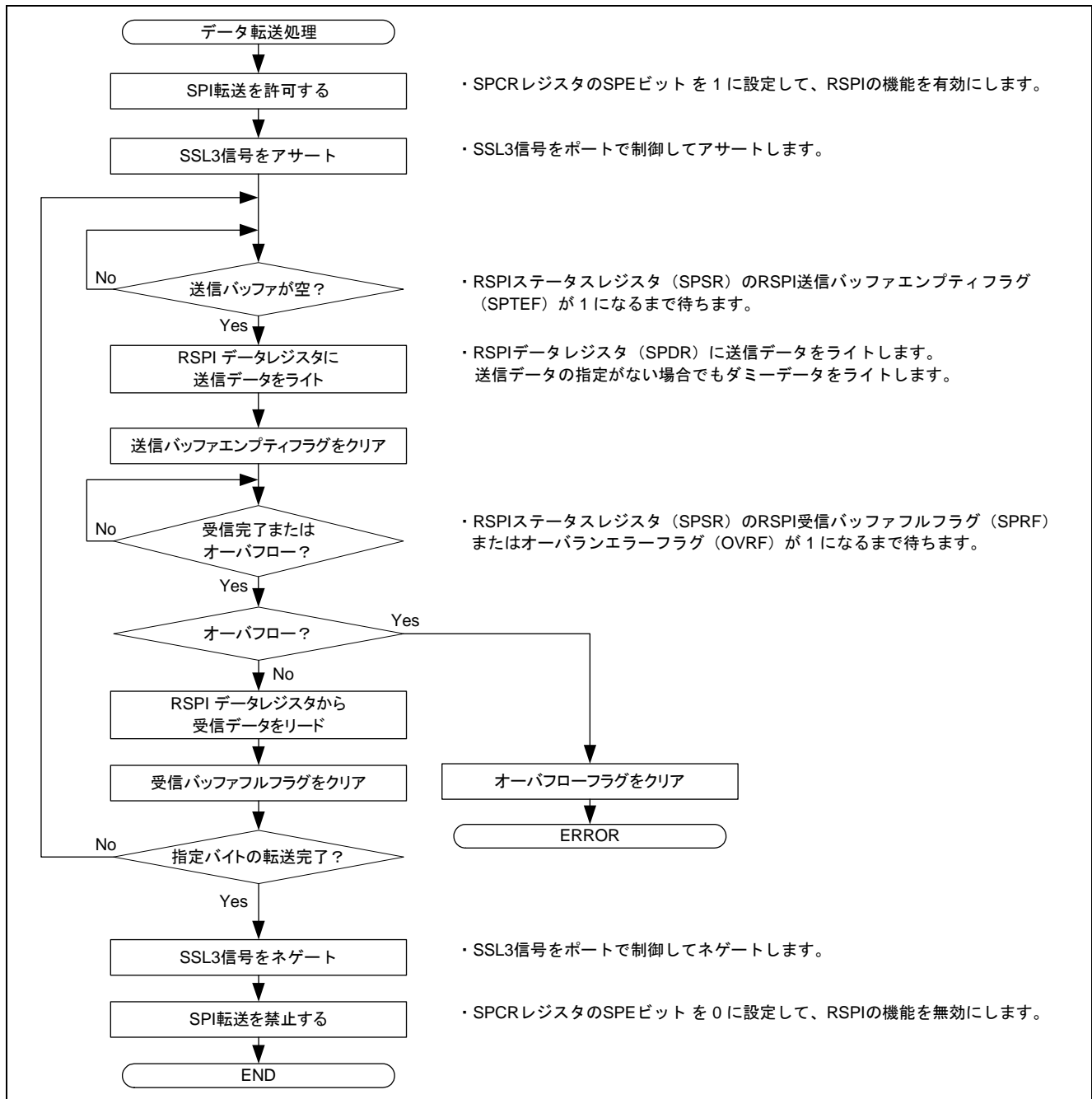


図15 参考プログラムのデータ転送手順フロー

3. 参考プログラム

ここでは、本参考プログラムの概要と MMC 操作関数 (API) について説明します。

3.1 概要

本参考プログラムは、SPI モードで MMC にアクセスします。アクセスの例として、ファイルシステムを想定したマスタブートレコード (MBR) の読み出しと、第 1 パーティションの読み出しを行っています。

SPI 転送には SH7216 が内蔵するルネサスシリアルペリフェラルインタフェースを使用します。転送方式の概要を以下に示します。

- 動作モード : モード 3 (アイドル時のクロックは'H'、偶数エッジでデータサンプリング)
- 転送速度 : 391kHz (MMC 認識時)、8.33MHz (MMC のデータ転送時)
- バッファ転送 : CPU 転送による 8 ビット幅アクセス
- 割り込み : 未使用

なお、MMC 制御に必要な処理は MMC 操作関数 (API) としてまとめています。

3.2 MMC 操作関数 (API) 一覧

表 5 に MMC 操作関数 (API) 一覧を示します。

表5 MMC 操作関数 (API) 一覧

関数名	説明
mmc_init_driver	MMC ドライバを初期化します。
mmc_attach	MMC アタッチ時にメディアの初期化処理を行います。
mmc_detach	MMC デタッチ時に終了処理を行います。
mmc_read_data	MMC からデータをリードします。
mmc_write_data	MMC にデータをライトします。
mmc_check_card	MMC の挿入を検出します。
mmc_get_info	カード情報を取得します。

3.3 マクロ（定数）一覧

表 6に MMC 操作関数（API）で使用するマクロ一覧を示します。

表6 MMC 操作関数（API）で使用するマクロ定義一覧

分類	マクロ名	値	説明
スロット管理	MMC_SLOT_NUM	1	対応するスロット数
	MMC_SLOT0	0	スロット番号 0
	MMC_SLOT1	1	スロット番号 1
エラーコード	MMC_OK	0	成功
	MMC_ERR_PARAM	-1	パラメータエラー
	MMC_ERR_HARD	-2	ハードウェアエラー
	MMC_ERR_CRC	-3	CRC エラー
	MMC_ERR_WP	-4	ライトプロテクトエラー
	MMC_ERR_MBLKCMD	-5	マルチブロックコマンドエラー
	MMC_ERR_IDLE	-6	アイドルステートエラー
	MMC_ERR_OTHER	-7	その他のエラー
フラグ管理	MMC_TRUE	0x01	フラグ ON
	MMC_FALSE	0x00	フラグ OFF
カードタイプ	MMC_CARD_UNDETECT	0x00	カード未検出
	MMC_CARD_MMC	0x01	MMC 検出
	MMC_CARD_OTHER	0xFF	その他のカードを検出
ライトプロテクト	MMC_NO_PROTECT	0x00	ライトプロテクトなし
	MMC_W_PROTECT_HARD	0x01	ハードウェアによるライトプロテクト
	MMC_W_PROTECT_SOFT	0x02	ソフトウェアによるライトプロテクト
データ転送モード	MMC_MODE_NORMAL	0x00	ノーマル転送モード
	MMC_MODE_DIRECT	0x01	直接転送モード（未使用）
	MMC_MODE_FORCED_W	0x02	強制書き込みモード（ライトプロテクト無効）
カード情報	MMC_BLK_SIZE	512	ブロックサイズ
	MMC_CRC_SIZE	2	CRC サイズ
	MMC_CSD_SIZE	16	CSD レジスタサイズ
	MMC_CID_SIZE	16	CID レジスタサイズ
	MMC_OCR_SIZE	4	OCR レジスタサイズ

3.4 構造体一覧

図 16に MMC 操作関数 (API) で使用する構造体一覧を示します。

■MMC 情報を定義する構造体

```
typedef struct {
    unsigned char Card;          /* Card type          */
    unsigned char WProtect;     /* Write-protection status */
    unsigned long MemSize;      /* Card capacity      */
    unsigned long MaxBlkNum;    /* The number of the max blocks */
} MMC_INFO;                   /* total 10byte      */
```

■CSD 情報を定義する構造体

```
typedef struct {
    unsigned short Taac;        /* Asynchronous data access time */
    unsigned short Nsac;       /* Synchronization data access time */
    int Nac;                    /* Data access time          */
    unsigned long MemSize;      /* Card capacity            */
    unsigned long MaxBlkNum;    /* The number of the max blocks */
    unsigned char WP;           /* Write-protection information */
    unsigned char Reserve[3];
} CSD_INFO;                   /* total 20byte        */
```

図16 MMC 操作関数 (API) で使用する構造体一覧

3.5 変数一覧

表 7に MMC 操作関数 (API) で使用する変数一覧を示します。これらの変数は接続されている MMC のステータスを確認するために使用できます。変数の値は MMC 操作関数が格納します。

表7 MMC 操作関数 (API) で使用する変数一覧

宣言	説明
unsigned char gMmc_Media [MMC_SLOT_NUM];	検出したカードのタイプを格納します。使用するマクロ定義については表 6のカードタイプを参照してください。
unsigned char gMmc_WP [MMC_SLOT_NUM];	ライトプロテクトの状態を格納します。使用するマクロ定義については表 6のライトプロテクトを参照してください。
unsigned short gMmc_AddrRev [MMC_SLOT_NUM];	ブロックサイズ (バイト) を格納します。mmc_data_read 関数と mmc_data_write 関数では、ブロック番号からアドレス値を算出するために使用しています。
CSD_INFO gMmc_CsdInfo [MMC_SLOT_NUM];	カードアタッチ時に読み出した CSD レジスタ情報を CSD_INFO 構造体に格納します。
unsigned char gMmc_CsdBuf [MMC_SLOT_NUM][MMC_CSD_SIZE];	CSD レジスタを読み出すためのバッファ領域です。CSD レジスタはカードアタッチ時に読み出しています。
unsigned char gMmc_CidBuf [MMC_SLOT_NUM][MMC_CID_SIZE];	CID レジスタを読み出すためのバッファ領域です。CID レジスタはカードアタッチ時に読み出しています。
unsigned char gMmc_OcrBuf [MMC_SLOT_NUM][MMC_OCR_SIZE];	OCR レジスタを読み出すためのバッファ領域です。OCR レジスタはカードアタッチ時に読み出しています。

3.6 関数仕様

ここではメイン関数と MMC 操作関数 (API) の関数仕様を示します。

main

ユーザ関数

メイン関数

書式 `void main(void);`

戻り値 なし

解説 ハードウェアおよびドライバの変数を初期化します。
 スロットに MMC が挿入されるまで待ち、挿入されたら MMC を初期化します。
 初期化後、マスタブートレコード (MBR) と第 1 パーティションの先頭セクタを読み出します。

mmc_init_driver

MMC 操作関数 (API)

MMC ドライバの初期化

書式 `void mmc_init_driver(void);`

戻り値 なし

解説 MMC ドライバを初期化します。
 — 制御モジュール (ルネサスシリアルペリフェラルインタフェース) の初期化
 — すべてのスロットの起動プロセスを実行
 (1) カード制御に必要な端子設定
 (2) ドライバ変数領域の初期化
 本関数は、システム起動後に一回実行します。

mmc_attach

MMC 操作関数 (API)

MMC スロット初期化関数 (アタッチ処理)

書式	<pre>int mmc_attach(int slot_no); int slot_no スロット番号</pre>												
戻り値	<table><tr><td>MMC_OK</td><td>成功</td></tr><tr><td>MMC_ERR_PARAM</td><td>パラメータエラー</td></tr><tr><td>MMC_ERR_HARD</td><td>ハードウェアエラー</td></tr><tr><td>MMC_ERR_CRC</td><td>CRC エラー</td></tr><tr><td>MMC_ERR_IDLE</td><td>アイドルステートエラー</td></tr><tr><td>MMC_ERR_OTHER</td><td>その他のエラー (カード未検出)</td></tr></table>	MMC_OK	成功	MMC_ERR_PARAM	パラメータエラー	MMC_ERR_HARD	ハードウェアエラー	MMC_ERR_CRC	CRC エラー	MMC_ERR_IDLE	アイドルステートエラー	MMC_ERR_OTHER	その他のエラー (カード未検出)
MMC_OK	成功												
MMC_ERR_PARAM	パラメータエラー												
MMC_ERR_HARD	ハードウェアエラー												
MMC_ERR_CRC	CRC エラー												
MMC_ERR_IDLE	アイドルステートエラー												
MMC_ERR_OTHER	その他のエラー (カード未検出)												
解説	<p>MMC のカードスロットを初期化します。</p> <ul style="list-style-type: none">— カード管理変数の初期化— カードの初期化 <p>本関数は、カードの挿入を検出した際に実行します。</p>												

mmc_detach

MMC 操作関数 (API)

MMC スロットデタッチ関数

書式	<pre>int mmc_detach(int slot_no); int slot_no スロット番号</pre>				
戻り値	<table><tr><td>MMC_OK</td><td>成功</td></tr><tr><td>MMC_ERR_PARAM</td><td>パラメータエラー</td></tr></table>	MMC_OK	成功	MMC_ERR_PARAM	パラメータエラー
MMC_OK	成功				
MMC_ERR_PARAM	パラメータエラー				
解説	<p>指定したスロットに対してデタッチ処理を実行します。</p> <ul style="list-style-type: none">— 制御モジュール (ルネサスシリアルペリフェラルインタフェース) の初期化— カード制御に必要な端子設定— カード管理変数の初期化 <p>本関数は、カードが抜かれた際に実行します。</p>				

mmc_read_data

MMC 操作関数 (API)

データリード関数

書式

```
int mmc_read_data(int slot_no, unsigned long blkno, unsigned long blkcnt,
unsigned char * buff, int mode);
```

int slot_no		スロット番号
unsigned long blkno		リードを開始するブロックのブロック番号
unsigned long blkcnt		リードするブロック数
unsigned char * buff	O	リードデータを格納するバッファ
int mode		データ転送モード

戻り値

MMC_OK	成功
MMC_ERR_PARAM	パラメータエラー
MMC_ERR_HARD	ハードウェアエラー
MMC_ERR_CRC	CRC エラー
MMC_ERR_OTHER	その他のエラー (カード未検出)

解説

ブロック単位 (512 バイト) でカードからデータを読み出します。
指定のブロック番号から指定のブロック数を読み出します。
以下のデータ転送モードで、データ転送方式を選択します。

- MMC_MODE_NORMAL : リードデータは引数で指定したバッファ領域に格納

mmc_write_data

MMC 操作関数 (API)

データライト関数

書式

```
int mmc_write_data(int slot_no, unsigned long blkno, unsigned long blkcnt,
unsigned char * buff, int mode);
```

int slot_no		スロット番号
unsigned long blkno		ライトを開始するブロックのブロック番号
unsigned long blkcnt		ライトするブロック数
unsigned char * buff		ライトデータを格納するバッファ
int mode		データ転送モード

戻り値

MMC_OK	成功
MMC_ERR_PARAM	パラメータエラー
MMC_ERR_HARD	ハードウェアエラー
MMC_ERR_WP	ライトプロテクトエラー
MMC_ERR_OTHER	その他のエラー (カード未検出)

解説

ブロック単位 (512 バイト) でカードにデータを書き込みます。
指定のブロック番号から指定のブロック数を書き込みます。
以下のデータ転送モードで、データ転送方式を選択します。

- MMC_MODE_NORMAL : ライトデータは引数で指定したバッファ領域を使用
- MMC_MODE_FORCED_W : ライトプロテクトに関わらず、強制的に書き込む

mmc_check_card

MMC 操作関数 (API)

カード挿入のチェック関数

書式

```
int mmc_check_card(int slot_no, unsigned char* sts);
int slot_no        |   スロット番号
unsigned char* sts |   カード挿入状態を格納するバッファ
```

戻り値

MMC_OK	成功
MMC_ERR_PARAM	パラメータエラー

解説

カードの挿入状態をチェックして、ポートの状態を引数 sts に格納します。

- MMC_TRUE : カードを検出した
- MMC_FALSE : カード未検出

注意事項

端子のチャタリングは除去していません。必要に応じて呼び出し元で行ってください。

mmc_get_info

MMC 操作関数 (API)

カード情報取得関数

書式

```
int mmc_get_info(int slot_no, MMC_INFO *info);
int slot_no      |   スロット番号
MMC_INFO *info  |   O   カード情報を格納するバッファ
```

戻り値

MMC_OK	成功
MMC_ERR_PARAM	パラメータエラー

解説

カード情報 (カードタイプ、ライトプロテクト状態、容量、ブロック数) を返します。

4. 参考ドキュメント

- ソフトウェアマニュアル
SH-2A、SH2A-FPU ソフトウェアマニュアル Rev.3.00
(最新版をルネサスエレクトロニクスの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
SH7214 グループ、SH7216 グループ ユーザーズマニュアル ハードウェア編 Rev.2.00
(最新版をルネサスエレクトロニクスのホームページから入手してください。)
- MMC 規格書 MultiMediaCard (MMC) Electrical Standard, High Capacity Ver.4.2
URL: <http://www.jedec.org>

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。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.07.26	—	初版発行

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くと、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きを行ってください。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>