SH7216 Group

Accessing MultiMediaCard

Using the Renesas Serial Peripheral Interface

R01AN0039EJ0100
Rev. 1.00
Aug. 27. 2010

Introduction

This application note describes an example to access the MultiMediaCard (MMC) using the Renesas Serial Peripheral Interface.

Target Device

SH7216

Contents

1. Introduction

1.1 Specifications

This application accesses the MMC using the Renesas Serial Peripheral Interface.

1.2 Modules Used

- Renesas Serial Peripheral Interface
- Pin Function Controller

1.3 Applicable Conditions

MCU	SH7216
Operating Frequency	Internal clock: 200 MHz Bus clock: 50 MHz Peripheral clock: 50 MHz AD clock: 50 MHz
Integrated Development Environment	Renesas Electronics High-performance Embedded Workshop Ver.4.07.00
C Compiler	Renesas Electronics SuperH RISC engine Family C/C++ compiler package Ver.9.03 Release 00
Compiler Options	Default setting in the High-performance Embedded Workshop (-cpu=sh2afpu -fpu=single -debug -gbr=auto -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1)
MMC	Compliant with version 4.2 Note: MMC with version 4.3 or later does not support SPI mode.

1.4 Related Application Note

For more information, refer to the following application note:

- SH7216 Group Example of Initialization

1.5 About Active-low Pins (Signals)

The symbol "#" suffixed to the pin (or signal) names indicates that the pins (or signals) are active-low.

2. Applications

This application accesses the MMC using the Renesas Serial Peripheral Interface (RSPI). This section describes an overview of the MMC protocols and the RSPI. For more information about the MMC protocols, refer to the MMC specifications.

2.1 MMC Protocols Overview

2.1.1 SPI Mode

The MMC has two modes; MultiMediaCard mode (MMC mode) which is defined as its default operating mode, and SPI mode which is based on the SPI standard. This application uses SPI mode. As this application note assumes that the MMC is used in SPI mode, MMC mode is not described here.

2.1.2 Commands

Use the commands to access the MMC.

Figure 1 shows the command format. Fix bits start, host, and end to 0, 1, and 1, respectively. Set the command code as the number which is included in the command name. Set an argument to the command parameter according to the command. Set the CRC7 which is computed from 5 bytes including the command code and command parameter.

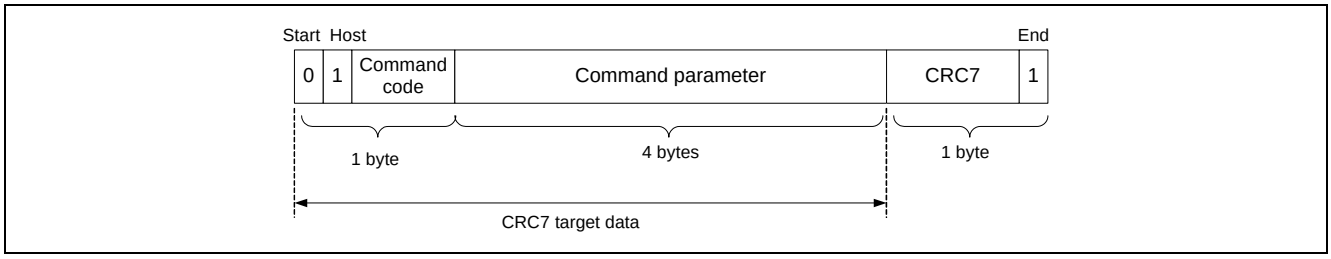


Figure 1 Command Format

Table 1 lists major commands.

Table 1 Major Commands

Name	Command Code	Description	Name	Command Code	Description
CMD0	H'00	Resets the MMC	CMD17	H'11	Reads the single block
CMD1	H'01	Initializes the MMC	CMD18	H'12	Reads the multiple blocks
CMD9	H'09	Reads the CSD register	CMD24	H'18	Writes a single block
CMD10	H'0A	Reads the CID register	CMD25	H'19	Writes the multiple blocks
CMD12	H'0C	Stops to read multiple blocks	CMD58	H'3A	Reads the OCR register
CMD13	H'0D	Reads the Status register	CMD59	H'3B	Turns the CRC ON or OFF

2.1.3 Responses

When the MMC receives a command from the SH7216, it returns a response. The response format varies depending on the command. This section describes R1 response as one of the major responses.

Figure 2 shows R1 response format.

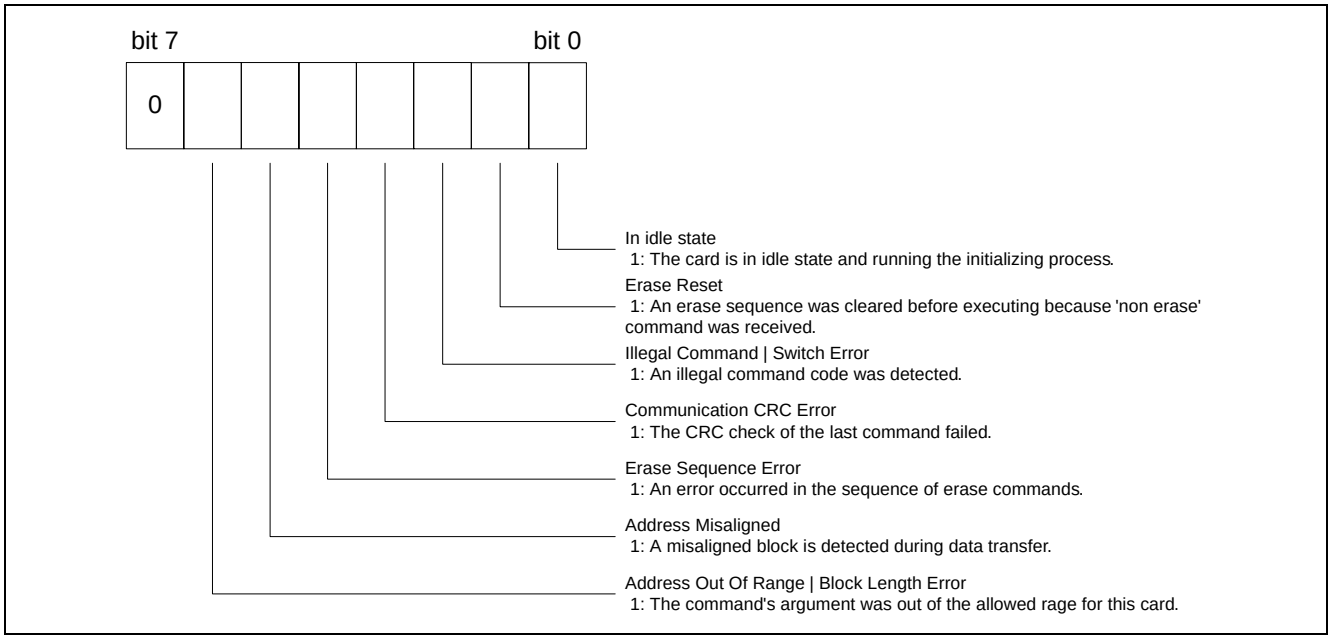


Figure 2 R1 Response

2.1.4 Data Token

When the MMC receives a command to read or write the data, the MMC (read command) or the SH7216 (write command) transfers the data using the data token after receiving the response.

Figure 3 shows the data token format. The MMC inserts the start data block token at the beginning of the data to distinguish an invalid value in idle state (H'FF) from the valid data H'FF. The MMC adds CRC16 at the end of the data, however, the start data block token is not included in the CRC calculation.

Table 2 lists the start data block tokens.

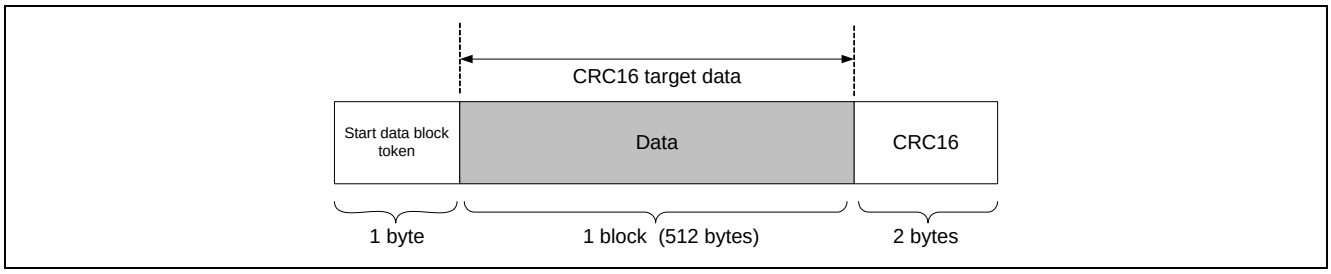


Figure 3 Data Token Format

Table 2 Start Data Block Tokens

Data Token Type	Value
Single/Multiple Block Read	H'FE (B'1111 1110)
Single Block Write	H'FE (B'1111 1110)
Multiple Block Write (Start)	H'FC (B'1111 1100)
Multiple Block Write (Stop)	H'FD (B'1111 1101)

Figure 4 shows the data error token format. If the MMC cannot transfer the data when the SH7216 reads the data, the MMC returns the data error token to the SH7216, instead of the data token.

When the SH7216 writes data in the MMC, the MMC returns the data response to the SH7216 to indicate whether the MMC writes the data successfully.

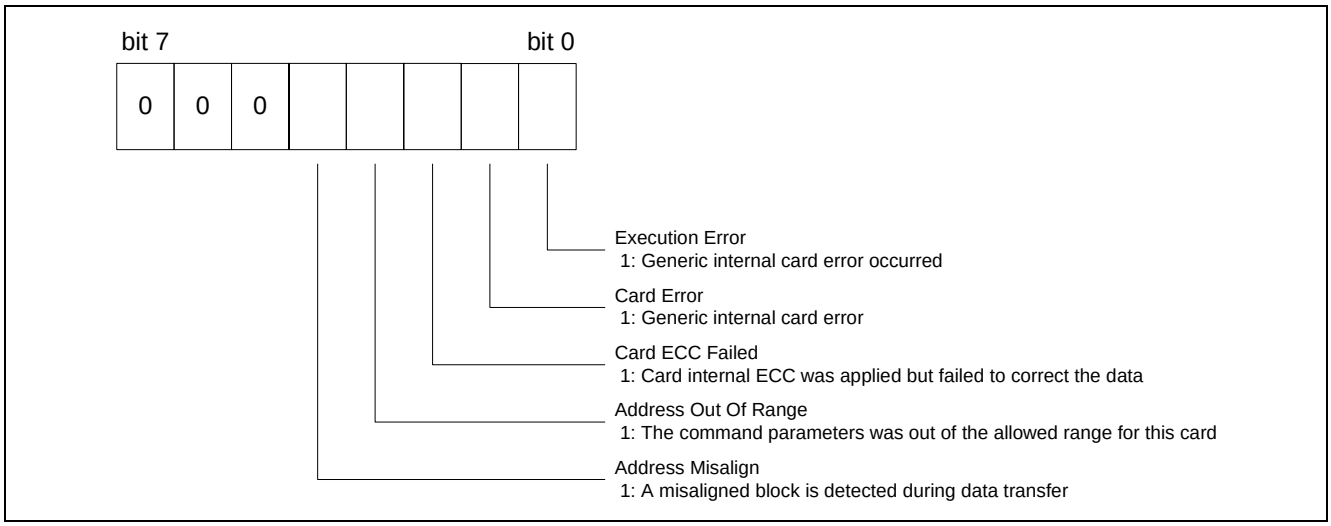


Figure 4 Data Error Token Format

2.1.5 MMC Access Sequence

This section describes the MMC access sequence.

Figure 5 shows the MMC access sequence when there is no data. The SH7216 issues the command, and then receives the response. If the SH7216 tries to receive data when the MMC is in idle state, the SH7216 receives H'FF data. Discard the received H'FF data. When the MMC is busy (H'00), wait until it receives H'FF data (the MMC is in idle state).

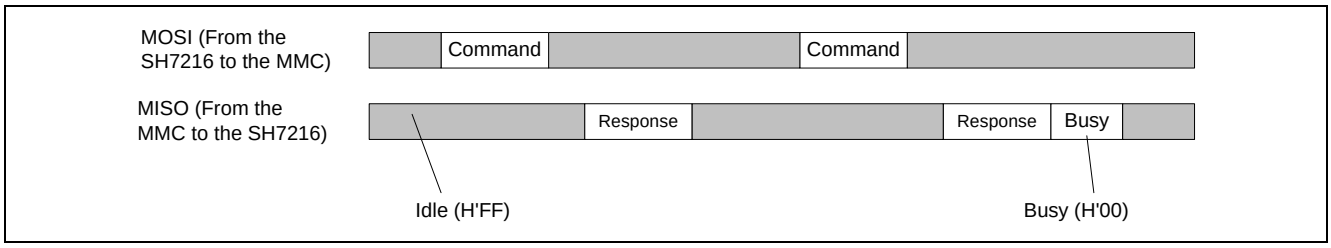


Figure 5 Access Sequence with No Data

Figure 6 shows the data read sequence when reading multiple blocks. The SH7216 issues the Multiple Block Read command, receives a response, and then a data token. The SH7216 issues the Stop command to stop the communication.

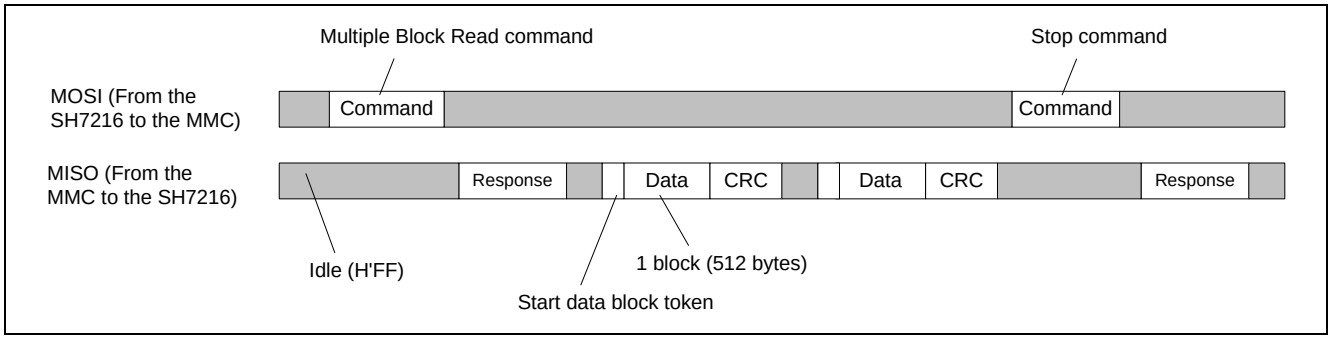


Figure 6 Data Read Sequence

Figure 7 shows the data write sequence when writing multiple blocks. The SH7216 issues the Multiple Block Write command, receives a response, and then transmits a data token. After transmitting a data token, the SH7216 receives a data response. The SH7216 issues the Multiple Block Write (Stop) data token to stop the communication.

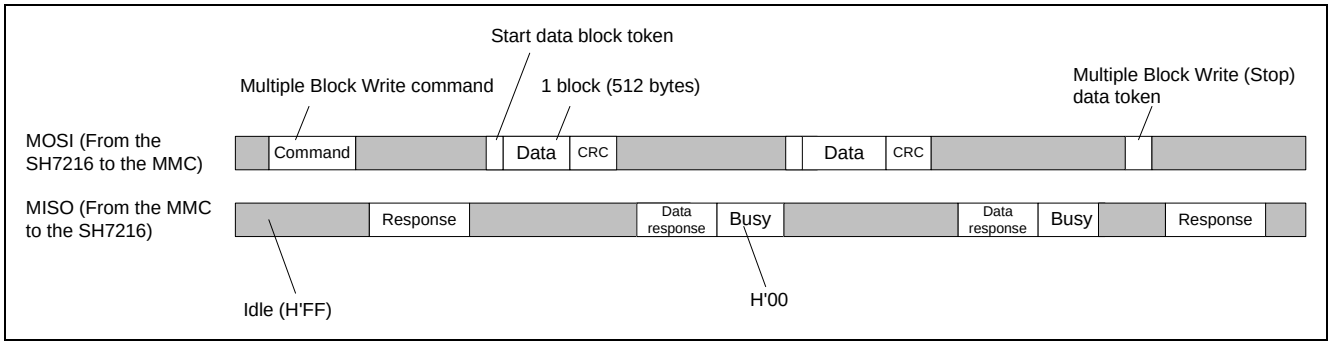


Figure 7 Data Write Sequence

2.1.6 CRC

The MMC detects an error using CRC7 and CRC16. Figure 8 and Figure 9 show CRC7 generator/checker and CRC16 generator/checker. CRC is turned OFF by default in SPI mode, however, use CMD59 to turn ON or OFF the CRC. Even when the CRC is turned OFF, transmitting or receiving CRC cannot be omitted.

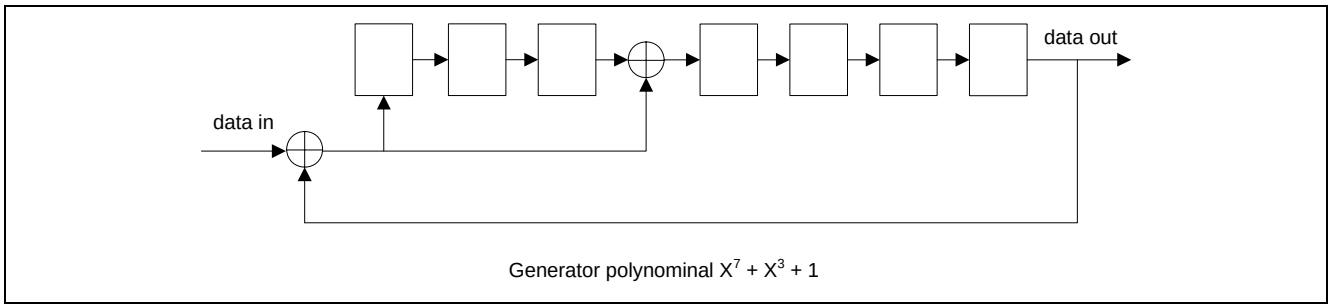


Figure 8 CRC7 Generator/Checker

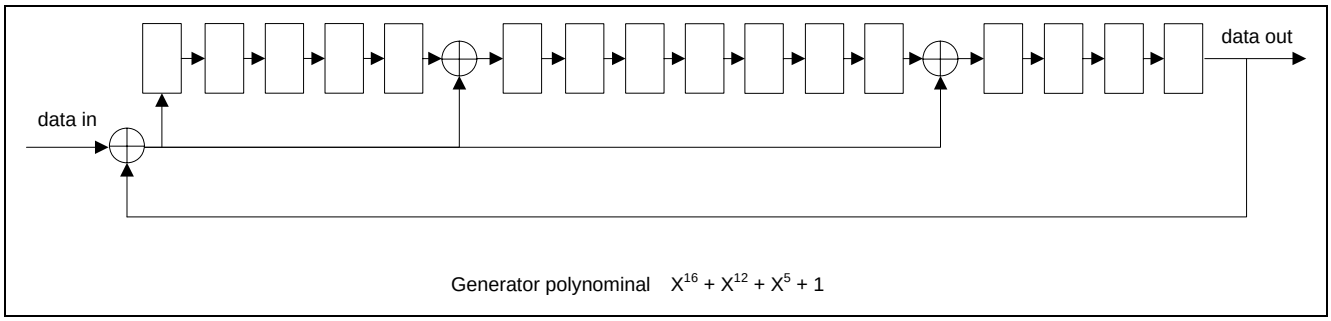


Figure 9 CRC16 Generator/Checker

2.1.7 Initializing the MMC

After power up, the MMC requires delay cycles as its initialization sequence. Then, assert the CS# signal, issue CMD0 to transition to SPI mode. Initialize the MMC by CMD1, and data transfer is enabled after the initialization sequence is completed.

Figure 10 shows the initialization steps when using the MMC in SPI mode.

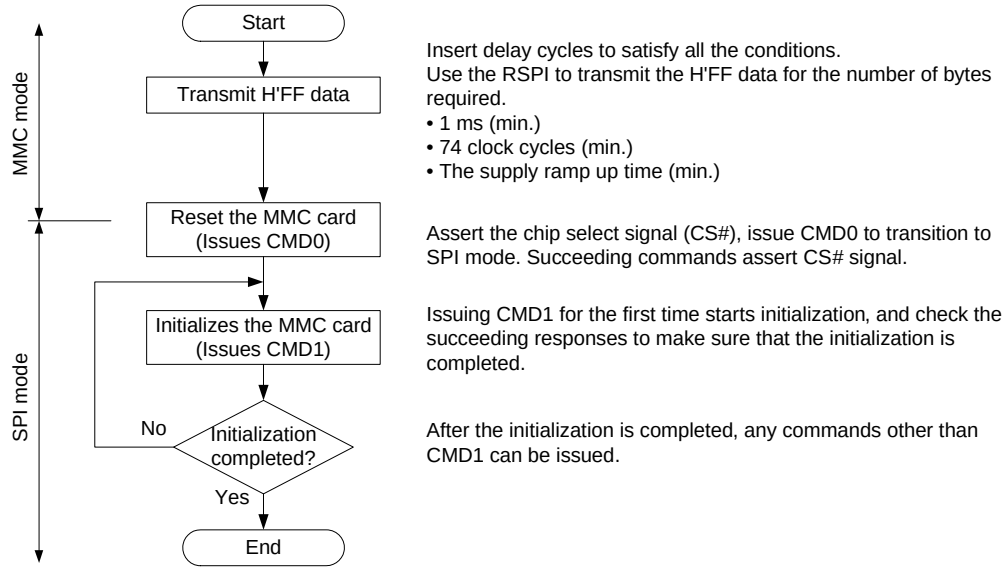


Figure 10 Initializing the MMC in SPI Mode

2.2 Renesas Serial Peripheral Interface (RSPI)

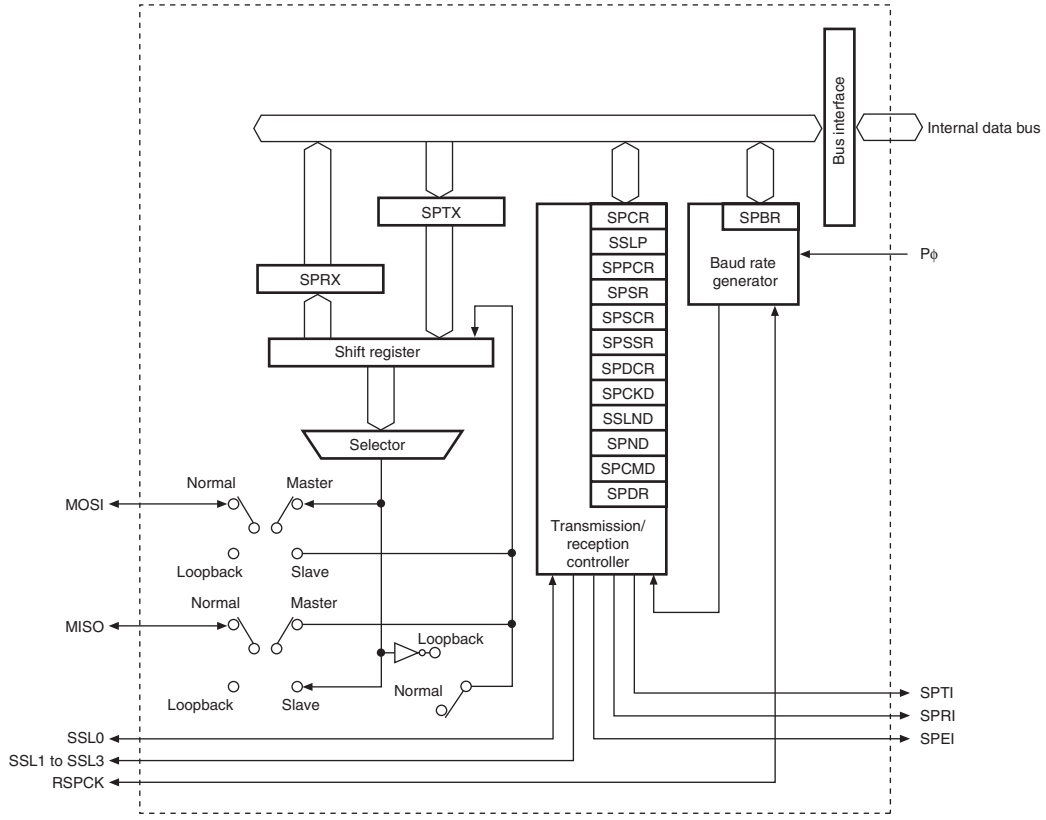
This section describes how to use the SH7216 Renesas Serial Peripheral Interface (RSPI).

2.2.1 RSPI Operation

The SH7216 includes one channel of the Renesas Serial Peripheral Interface (RSPI). It allows the serial communication with peripheral devices in 4-wire SPI operation and 3-wire clock synchronous operation using pins MOSI (Master Out Slave In), MISO (Master In Slave Out), SSL (Slave Select), and RSPCK (SPI Clock). As the RSPI has the following features, it supports various SPI-compliant devices:

- Master/slave modes
- Serial transfer clock with programmable polarity and phase (change SPI mode)
- Transfer bit length selectable from 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 24, and 32

Figure 11 shows the RSPI block diagram.



[Legend]
SPCR: RSPI control register
SSLP: RSPI slave select polarity register
SPPCR: RSPI pin control register
SPSR: RSPI status register
SPSCR: RSPI sequence control register
SPSSR: RSPI sequence status register
SPDCR: RSPI data control register
SPCKD: RSPI clock delay register
SSLND: RSPI slave select negate delay register
SPND: RSPI next-access delay register
SPCMD: RSPI command registers 0 to 3
SPBR: RSPI bit rate register
SPTX: RSPI transmit buffer
SPRX: RSPI receive buffer
SPTI: RSPI transmit interrupt
SPRI: RSPI receive interrupt
SPEI: RSPI error interrupt
SPDR: RSPI data register

Figure 11 RSPI Block Diagram

2.2.2 MMC Card Connections to the SH7216

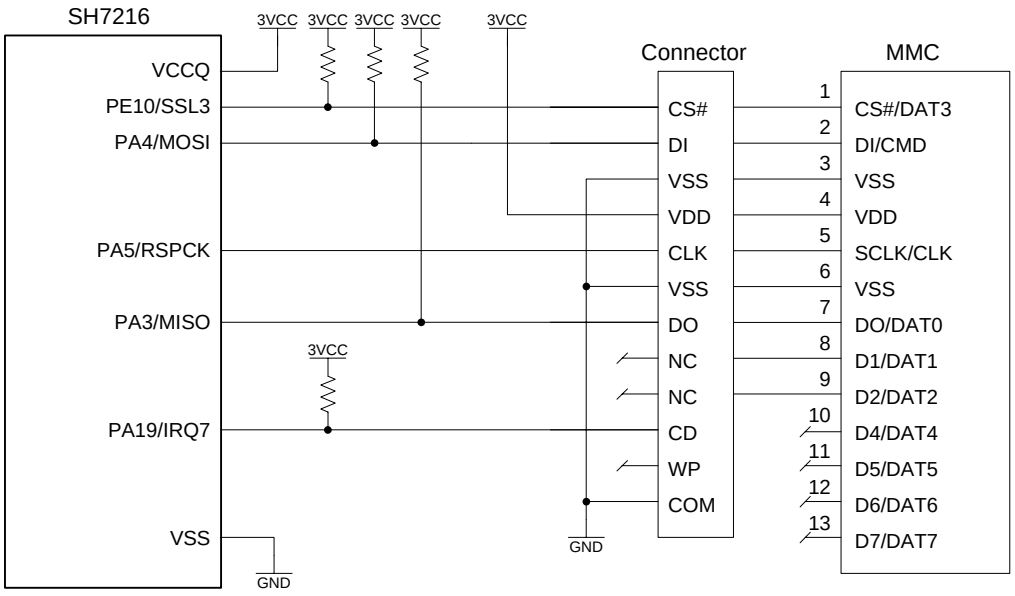
Table 3 lists the MMC pins. Pin configurations for MMC mode and SPI mode are different, and the MMC is in MMC Mode when it is powered on. As the MMC uses only common pins with SPI and MMC modes before transition to SPI mode, the MMC is allowed to communicate with pin connections in SPI mode.

Table 3 MMC Pins

Pin No.	SPI Mode		MMC Mode	
	Name	Description	Name	Description
1	CS#	Chip select	DAT3	Data
2	DI	Data IN	CMD	Command/response
3	VSS1	Ground	VSS1	Ground
4	VDD	Power supply	VDD	Power supply
5	SCLK	Clock	CLK	Clock
6	VSS2	Ground	VSS2	Ground
7	DO	Data OUT	DAT0	Data
8	Not used	–	DAT1	Data
9	Not used	–	DAT2	Data
10	Not used	–	DAT4	Data
11	Not used	–	DAT5	Data
12	Not used	–	DAT6	Data
13	Not used	–	DAT7	Data

Figure 12 shows the MMC connection in SPI mode. External pull-up resistors are installed on pins SSL3, MOSI, and MISO to avoid MMC malfunction when the MCU pins are high-impedance. The state of CD pin of the connector varies depending on insertion or removal of the MMC. The SH7216 detects the MMC insertion using this signal, however, note that the example shown in Figure 12 does not support "hot-insertion and removal".

Set the SH7216 pins according to the Pin Function Controller (PFC) Settings listed in Table 4.



Note: This example does not support "hot-insertion and removal".

Figure 12 MMC Connection in SPI Mode

Table 4 Pin Function Controller (PFC) Settings

Register Name	Setting	Description
Port A control register H1 (PACRH1)	H'0000	Specify PA19 pin as general-purpose I/O ports (MMC insertion can also be detected using IRQ7, however, this example uses the input port to detect the insertion.)
Port A I/O register H (PAIORH)	H'0000	Specify PA19 pin as input
Port A control register L2 (PACRL2)	H'0055	Specify PA5 pin as RSPCK I/O (RSPI) Specify PA4 pin as MOSI I/O (RSPI)
Port A control register L1 (PACRL1)	H'5000	Specify PA3 pin as MISO I/O (RSPI)
Port E control register L3 (PECRL3)	H'0000	Specify PE10 pin as general-purpose I/O ports (This example controls the output port to generate the chip select signal, not using SSL pin function of the RSPI.)
Port E I/O register L (PEIORL)	H'0400	Specify PE10 pin as output

2.2.3 RSPI Configuration Procedure

Figure 13 and Figure 14 show flow charts for configuring the RSPI in the sample program. This setting enables the RSPI to operate in master mode.

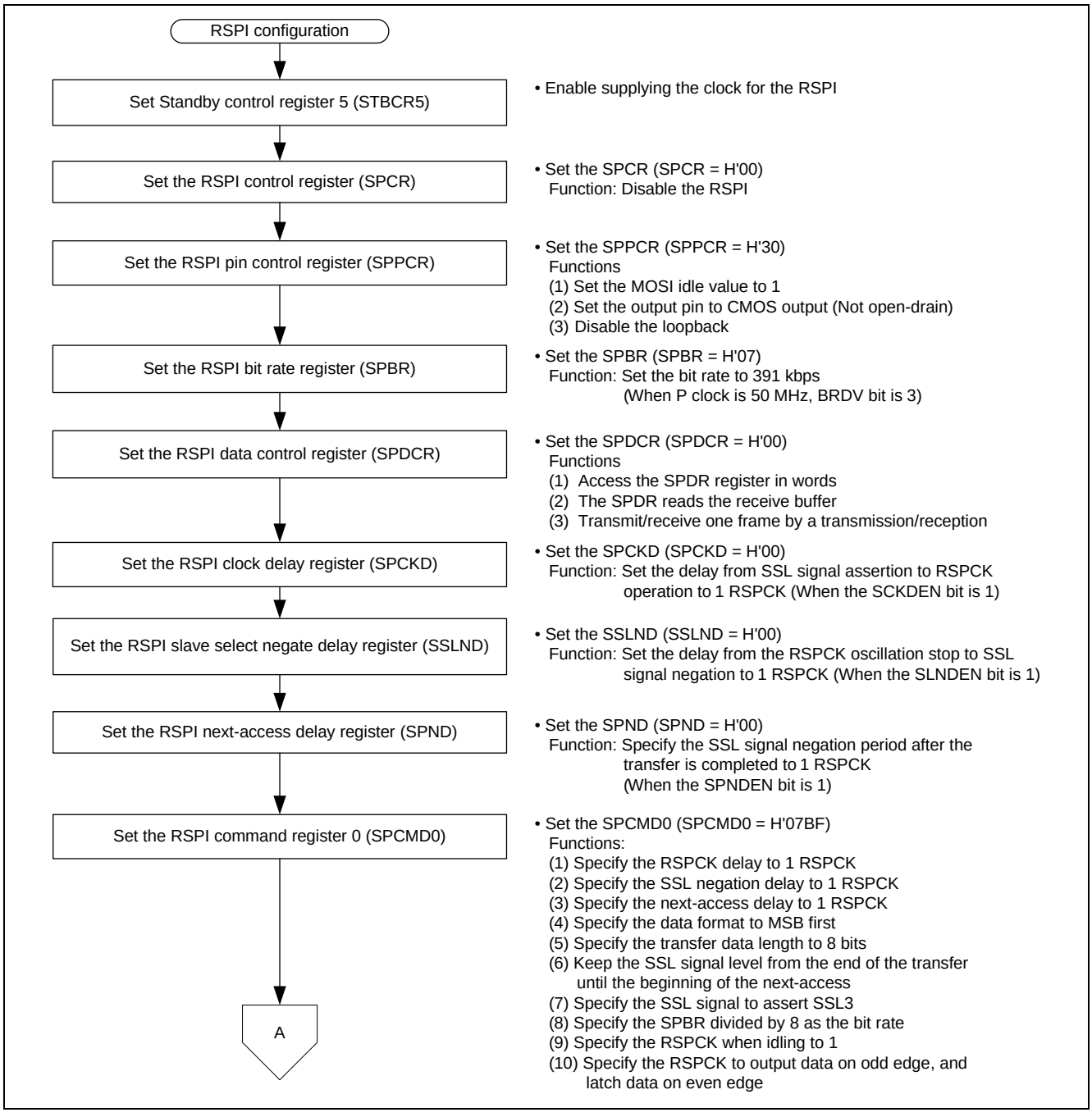


Figure 13 Flow Chart for Configuring the RSPI in the Sample Program (1/2)

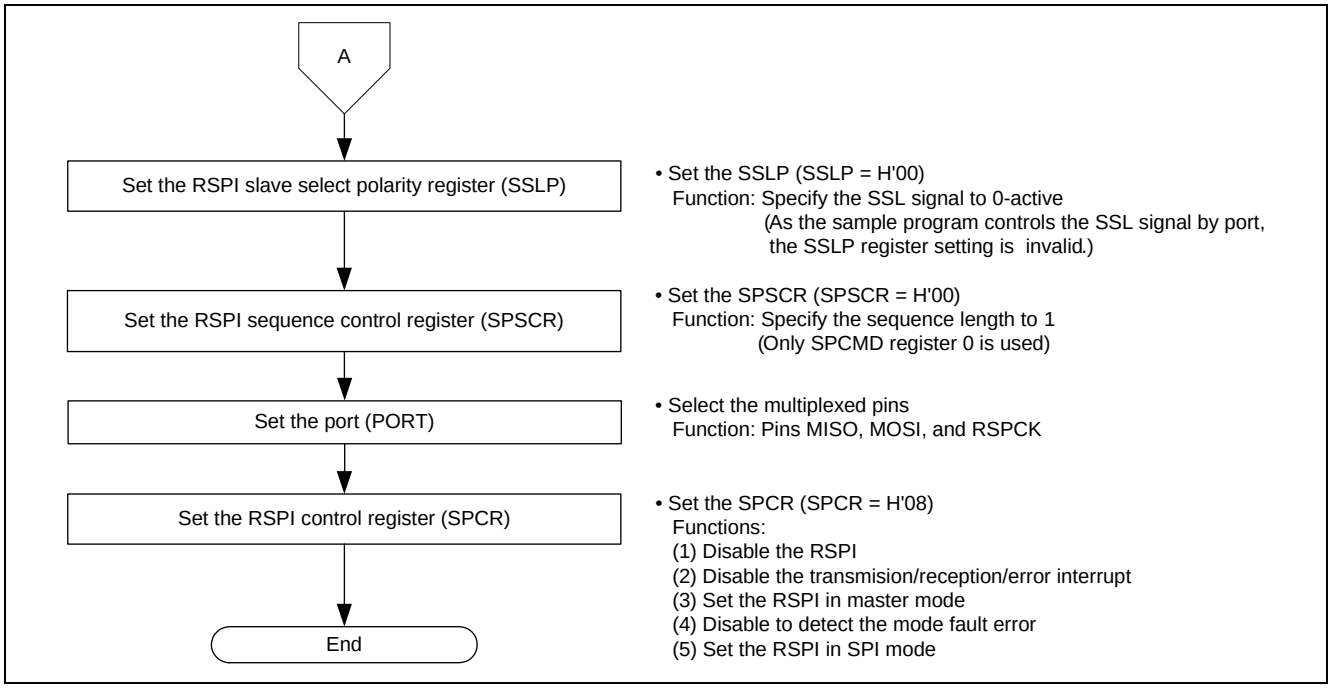


Figure 14 Flow Chart for Configuring the RSPI in the Sample Program (2/2)

2.2.4 RSPI Data Transfer Procedure

Figure 15 shows the flow chart for transferring data in the sample program. This setting allows the RSPI for full-duplex communication.

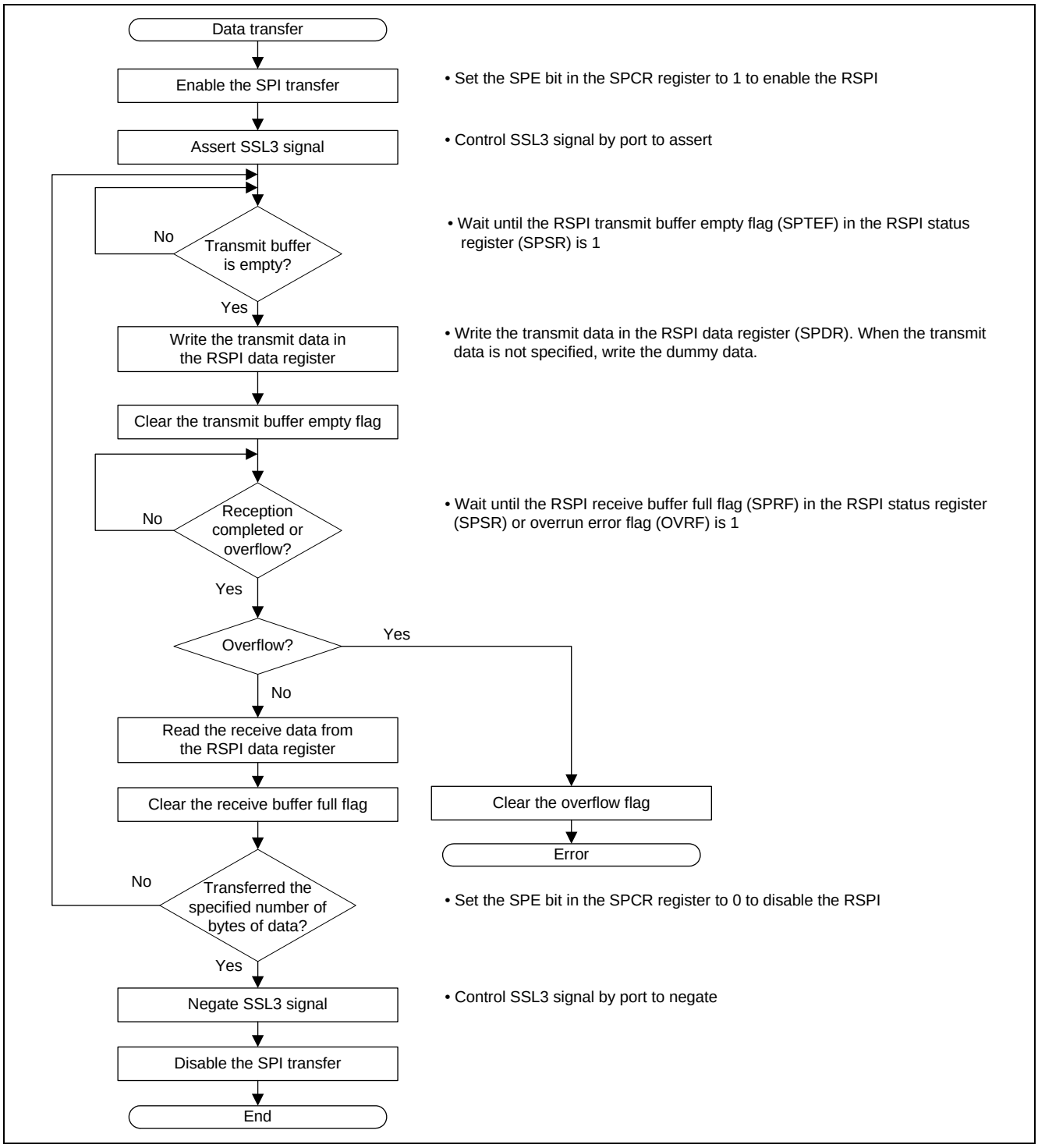


Figure 15 Flow Chart for Transferring Data in the Sample Program

3. Sample Program

This section describes an overview of the sample program and MMC manipulation function (API).

3.1 Overview

The sample program accesses the MMC in SPI mode. It reads the Master boot record (MBR) and the first partition, assuming that it is used in the file system.

Use the SH7216 Renesas Serial Peripheral Interface for the SPI transfer. The specifications of data transfer mode are as follows:

- Operating mode: Mode 3 (Clock while the RSPI is in idle state is 'H', and the RSPI latches the data on even edge)
- Transfer speed: 391 kHz (when the MMC is detected), 8.33 MHz (When the MMC is transferring data)
- Buffer transfer: Access in 8-bit wide by the CPU transfer
- Interrupt: Used

Processing required for controlling the MMC are described as MMC manipulation functions (API) in the following section.

3.2 MMC Manipulation Functions (API)

Table 5 lists the MMC manipulation functions (API).

Table 5 MMC Manipulation Functions (API)

Function Name	Description
mmc_init_driver	Initializes the MMC driver
mmc_attach	Initializes the media when the MMC is inserted
mmc_detach	Terminates the media when the MMC is removed
mmc_read_data	Reads the data from the MMC
mmc_write_data	Writes the data to the MMC
mmc_check_card	Detects the insertion of the MMC
mmc_get_info	Retrieves the card information

3.3 Macro Definitions (Constants)

Table 6 lists the macro definitions used in the MMC manipulation functions (API).

Table 6 Macro Definitions

Category	Macro Name	Value	Description
Slot Management	MMC_SLOT_NUM	1	Number of slots for the MMC
	MMC_SLOT0	0	Slot number 0
	MMC_SLOT1	1	Slot number 1
Error Code	MMC_OK	0	Successful
	MMC_ERR_PARAM	−1	Parameter error
	MMC_ERR_HARD	−2	Hardware error
	MMC_ERR_CRC	−3	CRC error
	MMC_ERR_WP	−4	Write-protect error
	MMC_ERR_MBLKCMD	−5	Multiple block command error
	MMC_ERR_IDLE	−6	Idle state error
	MMC_ERR_OTHER	−7	Other error
Flag Management	MMC_TRUE	0x01	Flag is ON
	MMC_FALSE	0x00	Flag is OFF
Card Type	MMC_CARD_UNDETECT	0x00	The card is not detected
	MMC_CARD_MMC	0x01	The MMC is detected
	MMC_CARD_OTHER	0xFF	The other type of card is detected
Write-protection	MMC_NO_PROTECT	0x00	Not write-protected
	MMC_W_PROTECT_HARD	0x01	Write-protected by hardware
	MMC_W_PROTECT_SOFT	0x02	Write-protected by software
Data Transfer Mode	MMC_MODE_NORMAL	0x00	Normal transfer mode
	MMC_MODE_DIRECT	0x01	Direct transfer mode (Not used)
	MMC_MODE_FORCED_W	0x02	Force writing mode (Write-protection is invalid)
Card Information	MMC_BLK_SIZE	512	Block size
	MMC_CRC_SIZE	2	CRC size
	MMC_CSD_SIZE	16	CSD register size
	MMC_CID_SIZE	16	CID register size
	MMC_OCR_SIZE	4	OCR register size

3.4 Structures

Figure 16 shows the structures used in the MMC manipulation functions (API).

- Structures to define the MMC information

```
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      */
```

- Structures to define the CSD information

```
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         */
```

Figure 16 Structures Used in the MMC Manipulation Functions (API)

3.5 Variables

Table 7 lists the variables used in the MMC manipulation functions (API). Use these constants to make sure the status of the MMC inserted. The MMC manipulation functions store the variable value.

Table 7 Variables Used in the MMC Manipulation Functions (API)

Declaration	Description
unsigned char gMmc_Media [MMC_SLOT_NUM];	Stores the type of the card detected. For the macro definition used, refer to Card Type in Table 6.
unsigned char gMmc_WP [MMC_SLOT_NUM];	Stores the status of the write-protection. For the macro definition used, refer to Write-protection in Table 6.
unsigned short gMmc_AddrRev [MMC_SLOT_NUM];	Stores the block size (in bytes). Functions mmc_data_read and mmc_data_write use this variable to calculate the address from the block number.
CSD_INFO gMmc_CsdInfo [MMC_SLOT_NUM];	Reads the CSD register information when the card is inserted, and stores the information in the CSD_INFO structure.
unsigned char gMmc_CsdBuf [MMC_SLOT_NUM] [MMC_CSD_SIZE];	This is the buffer to read the CSD register, which is read when the card is inserted.
unsigned char gMmc_CidBuf [MMC_SLOT_NUM] [MMC_CID_SIZE];	This is the buffer to read the CID register, which is read when the card is inserted.
unsigned char gMmc_OcrBuf [MMC_SLOT_NUM] [MMC_OCR_SIZE];	This is the buffer to read the OCR register, which is read when the card is inserted.

3.6 Functions

This section describes the main function and MMC manipulation functions (API).

main

User function

Main function

Format `void main(void);`

Return value None

Description Initializes the hardware and driver variables
Wait until the MMC is inserted, and initialize the MMC.
After initialization, it reads the Master boot record (MBR) and the starting sector of the first partition.

mmc_init_driver

MMC manipulation function (API)

Initializes the MMC driver

Format `void mmc_init_driver(void);`

Return value None

Description Initializes the MMC driver
- Initializes the control module (Renesas Serial Peripheral Interface)
- Starts up all slots
  (1) Sets the pins to control the card
  (2) Initializes the driver variable area

Execute this function once after the system is powered on.

mmc_attach

MMC manipulation function (API)

Initializes the MMC slot (attach process)

Format
```
int mmc_attach(int slot_no);
   int slot_no          I    Slot number
```

Return value

MMC_OK	Successful
MMC_ERR_PARAM	Parameter error
MMC_ERR_HARD	Hardware error
MMC_ERR_CRC	CRC error
MMC_ERR_IDLE	Idle state error
MMC_ERR_OTHER	Other error (Card is not detected)

Description

Initialize the MMC card slot
- Initializes the card control variables
- Initializes the card

Execute this function when an insertion of the card is detected.

mmc_detach

MMC manipulation function (API)

Detaches the MMC slot

Format
```
int mmc_detach(int slot_no);
   int slot_no          I    Slot number
```

Return value

MMC_OK	Successful
MMC_ERR_PARAM	Parameter error

Description

Executes the detach process to the specified slot
- Initializes the control module (Renesas Serial Peripheral Interface)
- Sets the pins to control the card
- Initializes the card control variables

Execute this function when the card is removed.

mmc_read_data

MMC manipulation function (API)

Reads the data from the card

Format

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

`int slot_no`	I	Slot number
`unsigned long blkno`	I	Block number to start reading the data
`unsigned long blkcnt`	I	Number of blocks to read
`unsigned char * buff`	O	Buffer to store the read data
`int mode`	I	Data transfer mode

Return value

MMC_OK	Successful
MMC_ERR_PARAM	Parameter error
MMC_ERR_HARD	Hardware error
MMC_ERR_CRC	CRC error
MMC_ERR_OTHER	Other error (Card is not detected)

Description

Reads the data from the card in blocks (512 bytes).
Reads the specified number of blocks from the specified block number.
Specify the data transfer mode by the following mode.

- MMC_MODE_NORMAL: Store the read data in the buffer area specified by an argument

mmc_write_data

MMC manipulation function (API)

Writes the data to the card

Format

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

`int slot_no`	I	Slot number
`unsigned long blkno`	I	Block number to start writing the data
`unsigned long blkcnt`	I	Number of blocks to write
`unsigned char * buff`	I	Buffer to store the write data
`int mode`	I	Data transfer mode

Return value

MMC_OK	Successful
MMC_ERR_PARAM	Parameter error
MMC_ERR_HARD	Hardware error
MMC_ERR_WP	Write-protect error
MMC_ERR_OTHER	Other error (Card is not detected)

Description

Writes the data to the card in blocks (512 bytes).
Writes the specified number of blocks from the specified block number.
Specify the data transfer mode by the following mode.

- MMC_MODE_NORMAL: Write data uses the buffer area specified by an argument
- MMC_MODE_FORCED_W: Force to write data even when the card is write-protected

mmc_check_card

MMC manipulation function (API)

Detects an insertion of a card

Format
```
int mmc_check_card(int slot_no, unsigned char* sts);
   int slot_no          I    Slot number
   unsigned char* sts   I    Buffer to store the card insertion state
```

Return value

MMC_OK	Successful
MMC_ERR_PARAM	Parameter error

Description
Detects the insertion of the card, and stores the port status in the argument *sts*
- MMC_TRUE: Card is detected
- MMC_FALSE: Card is not detected

Note
As the chattering on pins is not removed, remove any chattering on caller as appropriate.

mmc_get_info

MMC manipulation function (API)

Retrieves the card information

Format
```
int mmc_get_info(int slot_no, MMC_INFO *info);
   int slot_no          I    Slot number
   MMC_INFO *info       O    Buffer to store the card information
```

Return value

MMC_OK	Successful
MMC_ERR_PARAM	Parameter error

Description
Returns the card information (card type, write-protection status, capacity, and the number of blocks)

4. References

- Software Manual
  SH-2A/SH2A-FPU Software Manual Rev. 3.00
  The latest version of the software manual can be downloaded from the Renesas Electronics website.

- Hardware Manual
  SH7214 Group, SH7216 Group Hardware User's Manual Rev. 2.00

The latest version of the hardware user's manual can be downloaded from the Renesas Electronics website.

- MMC specifications MultiMediaCard (MMC) Electrical Standard, High Capacity Ver.4.2
  URL: http://www.jedec.org

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/inquiry

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Aug.27.10	—	First edition issued

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.
   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment when power is supplied.
   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.
   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.
5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
   - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
   - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
   - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
7F, No. 363 Fu Shing North Road Taipei, Taiwan, R.O.C.
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.0