Section 57. Secure Digital Host Controller (SDHC)

HIGHLIGHTS

This section of the manual contains the following major topics:

57.1 Introduction ................................................................. 57-2
57.2 Control Registers ............................................................. 57-3
57.3 SDHC Module Operation ...................................................... 57-29
57.4 Related Application Notes .............................................. 57-30
57.5 Revision History .............................................................. 57-31
57.1 INTRODUCTION

This family reference manual section describes the Secure Digital Host Controller (SDHC) module in the PIC32 family of devices.

The SDHC module uses a 32-bit System Bus master and slave interface to connect the Host system and standard card interface on the device side.

The core has a built-in DMA controller so that data can be automatically transferred between system memory and the SD/SDIO/eMMC card without intervention from the CPU.

The SDHC module includes the following features:

- SD Association (sdcard.org) specification compliance:
  - “SD Host Controller Simplified Specification” (version 2.00)
  - “Physical Layer Simplified Specification” (version 2.00)
  - “SDIO Simplified Specification” (version 2.00)
- Default and High-Speed modes of operation
- 1-bit or 4-bit data transfers
- Built-in clock divider
- PIO and ADMA modes of data transfer
- 3.3V operation
- Interrupt support
- Stop at block gap

A block diagram of the SDHC module is provided in Figure 57-1.

Figure 57-1: Secure Digital Host Controller (SDHC) Block Diagram
Section 57. Secure Digital Host Controller (SDHC)

57.2 CONTROL REGISTERS

SDHC operations are controlled using the following Special Function Registers (SFRs):

- **SDHCBCON**: SDHC Block Control Register
  This register is used to configure the number of bytes in a data block and the number of data blocks.

- **SDHCARG**: SDHC Argument Register
  This register contains the SD command argument.

- **SDHCMODE**: SDHC Mode Register
  This register is used to configure command and transfer modes.

- **SDHCRESPx**: SDHC Response Register ‘x’ (‘x’ = 0-3)
  These registers are used to store responses from Secure Digital (SD) cards.

- **SDHCDATA**: SDHC Data Register
  This register is used to access the internal buffer of the data port.

- **SDHCSTAT1**: SDHC Status Register 1
  This read-only register is used by the module driver to obtain the status of the Host Controller.

- **SDHCINTSTAT**: SDHC Interrupt Status Register
  This register is used by the module to report various normal and error interrupt status.

- **SDHCINTEN**: SDHC Interrupt Flag Enable Register
  This register is used to control individual normal and error interrupt flags.

- **SDHCINTSEN**: SDHC Interrupt Signal Enable Register
  This register is used to select which interrupt status is indicated to the Host System as the interrupt. These status bits all share the same 1-bit interrupt line. Setting any of these bits to ‘1’ enables interrupt generation.

- **SDHCSTAT2**: SDHC Status Register 2
  This register is used to indicate CMD12 response error of Auto CMD12.

- **SDHCCAP**: SDHC Capabilities Register
  This register specifies the supported features and performance capabilities of the Host Controller.

- **SDHCMAXCAP**: SDHC Maximum Current Capabilities Register
  This register indicates the maximum current capability for 3.3V signaling.

- **SDHCFE**: SDHC Force Event Register
  This register can be used to force events in the Interrupt Status Register. To generate an interrupt signal, both the Interrupt Signal Enable and Interrupt Flag Enable will be set.

- **SDHCADESTAT**: SDHC ADMA Error Status Register
  This register supports the ADMA Error Status Register as specified in the “SD Host Controller Simplified Specification” (version 2.00).

- **SDHCAADDR**: SDHC ADMA Address Register
  This register contains the physical descriptor address used for ADMA data transfers.
Table 57-1 provides a brief summary of the related SDHC module registers. Corresponding registers appear after the summary, followed by a detailed description of each bit.

<table>
<thead>
<tr>
<th>Register Name</th>
<th>Bit Range</th>
<th>Bit 31/15</th>
<th>Bit 30/14</th>
<th>Bit 29/13</th>
<th>Bit 28/12</th>
<th>Bit 27/21/10</th>
<th>Bit 23/7</th>
<th>Bit 22/6</th>
<th>Bit 21/5</th>
<th>Bit 20/4</th>
<th>Bit 19/3</th>
<th>Bit 18/2</th>
<th>Bit 17/1</th>
<th>Bit 16/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDHC BCON</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC ARG</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC MODE</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC RESPx</td>
<td>(x = 0-3)</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC DATA</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC STAT1</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC CON1</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SDHC CON2</td>
<td>31:16</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Legend: -- = unimplemented, read as '0'.
### Section 57. Secure Digital Host Controller (SDHC)

**Register 57-1: SDHCBCON: SDHC Block Control Register**

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>23:16</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>7:0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
</tbody>
</table>

**Legend:**
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- **x** = Bit is unknown

**bit 31-16** **BCOUNT<31:0>: Blocks Count for Current Transfer bits**

These bits represent the number of blocks. The software sets this value between 1 and 65,535 blocks and the SDHC decrements the count after each block transfer and stops when the count reaches zero.

- 0xFFFF = 65,535 blocks
- 0x0002 = 2 blocks
- 0x0001 = 1 block
- 0x0000 = Stop count

**bit 15-10** **Unimplemented:** Read as ‘0’

**bit 9-0** **BSIZE<9:0>: Transfer Block Size bits**

These bits specify the block size of the data transfer for CMD17, CMD18, CMD24, CMD25, and CMD53.

- 0x200 = 512 bytes
- 0x1FF = 511 bytes
- 0x002 = 2 bytes
- 0x001 = 1 byte
- 0x000 = No data transfer

**Note 1:** These bits are only used when the BCEN bit (SDHCMODE<1>) is set to ‘1’ and is valid only for multiple block transfers. The BCOUNT<15:0> bits need not be set if the BSIZE bit (SDHCMODE<5>) is set to ‘0’.

**2:** These bits can only be accessed when no transactions are in progress. Read operations during transfers will return an invalid value and write operations to these bits will be ignored.
### Register 57-2: SDHCARG: SDHC Argument Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td></td>
</tr>
<tr>
<td>23:16</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td></td>
</tr>
<tr>
<td>15:8</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td></td>
</tr>
<tr>
<td>7:0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- **x** = Bit is unknown

**bit 31-0**  **ARG<31:0>**: Command Argument bits
### Register 57-3: SDHCMODE: SDHC Mode Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 29/21/13/5</th>
<th>Bit 27/19/11/3</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CTYPE&lt;1:0&gt;</td>
<td>DPSEL</td>
<td>CIDXCEN&lt;2&gt;</td>
<td>CCRCCEN&lt;3&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BSEL</td>
<td>DTXDSEL</td>
<td>ACEN&lt;1:0&gt;</td>
<td>BCEN</td>
<td>DMAEN</td>
<td></td>
</tr>
<tr>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as '0'
- **-n** = Value at POR
- **'1'** = Bit is set
- **'0'** = Bit is cleared
- **x** = Bit is unknown

**bit 31-30 Unimplemented: Read as '0'**

**bit 29-24 CIDX<5:0>: Command Index bits**

These bits represent the command number (0-63).

**bit 23-22 CTYPE<1:0>: Command Type bits**

- 11 = Abort
- 10 = Resume
- 01 = Suspend
- 00 = Normal

**bit 21 DPSEL: Data Present Select bit**

- 1 = Data is present
- 0 = Data is not present

**bit 20 CIDXCEN: Command Index Check Enable bit**

- 1 = Command index check is enabled
- 0 = Command index check is disabled

**bit 19 CCRCCEN: Command CRC Check Enable bit**

- 1 = Command CRC check is enabled
- 0 = Command CRC check is disabled

**bit 18 Unimplemented: Read as '0'**

**bit 17-16 RESPTYPE<1:0>: Response Type Select bits**

- 11 = Response length 48; check busy after response
- 10 = Response length 48
- 01 = Response length 136
- 00 = No response

**bit 15-6 Unimplemented: Read as '0'**

**bit 5 BSEL: Multiple/Single Block Select bit**

- 1 = Multiple block, set when issuing multiple transfer commands using DAT lines
- 0 = Single block

**Note 1:** Refer to bits 45-40 of the command format in the “SD Host Controller Simplified Specification” (version 2.00).

**Note 2:** If these bits are set to ‘1’, the SDHC will check the index field in the response to see if it has the same value as the CIDX<5:0> bits, if not, it will be reported as a command index error.

**Note 3:** If these bits are set to ‘1’, the SDHC will check the CRC field in the response and reports a command CRC error upon a CRC error detection.
Register 57-3:  SDHCMODE: SDHC Mode Register (Continued)

bit 4  **DTXDSEL:** Data Transfer Direction Select bit
       1 = Read (card to SDHC)
       0 = Write (SDHC to card)

bit 3-2  **ACEN<1:0>:** Auto CMD12 Enable bits
        Auto CMD12 is used to stop multiple-block read/write operations.
        11 = Reserved
        10 = Reserved
        01 = Auto CMD12 is enabled
        00 = Auto CMD 12 is disabled

bit 1  **BCEN:** Block Count Enable Bit
       1 = Block count is enabled
       0 = Block count is disabled

bit 0  **DMAEN:** DMA Enable bit
       1 = DMA (ADMA) is used to transfer data
       0 = CPU is used to transfer data

**Note 1:** Refer to bits 45-40 of the command format in the “SD Host Controller Simplified Specification” (version 2.00).

2:  If these bits are set to ‘1’, the SDHC will check the index field in the response to see if it has the same value as the CIDX<5:0> bits, if not, it will be reported as a command index error.

3:  If these bits are set to ‘1’, the SDHC will check the CRC field in the response and reports a command CRC error upon a CRC error detection.
### Section 57. Secure Digital Host Controller (SDHC)

#### Register 57-4: SDHCRESPx: SDHC Response Register ‘x’ (’x’ = 0-3)

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
</tr>
<tr>
<td>23:16</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
</tr>
<tr>
<td>15:8</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
</tr>
<tr>
<td>7:0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
</tr>
</tbody>
</table>

**Legend:**

- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- **x** = Bit is unknown

**Table 57-2: Response Bit Definition for Each Response Type**

<table>
<thead>
<tr>
<th>Response Type (see Note 1)</th>
<th>Response Meaning</th>
<th>Response Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1, R1b (normal response)</td>
<td>Card status</td>
<td>SDHCRESP0&lt;31:0&gt;</td>
</tr>
<tr>
<td>R1b (Auto CMD12 response)</td>
<td>Card status for Auto CMD12</td>
<td>SDHCRESP3&lt;31:0&gt;</td>
</tr>
<tr>
<td>R2 (CID, CSD register)</td>
<td>CID or CSD register</td>
<td>SDHCRESP0&lt;31:0&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SDHCRESP1&lt;31:0&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SDHCRESP2&lt;31:0&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SDHCRESP3&lt;31:0&gt;</td>
</tr>
<tr>
<td>R3 (OCR register)</td>
<td>OCR register for memory</td>
<td>SDHCRESP0&lt;31:0&gt;</td>
</tr>
<tr>
<td>R4 (OCR register)</td>
<td>OCR register for I/O, etc.</td>
<td>SDHCRESP0&lt;31:0&gt;</td>
</tr>
<tr>
<td>R5, R5b</td>
<td>SDIO response</td>
<td>SDHCRESP0&lt;31:0&gt;</td>
</tr>
<tr>
<td>R6 (published RCA response)</td>
<td>New published RCA&lt;31:16&gt;, etc.</td>
<td>SDHCRESP0&lt;31:0&gt;</td>
</tr>
</tbody>
</table>

**Note 1:** For additional information, refer to the “SD Host Controller Simplified Specification” (version 2.00), the “Physical Layer Simplified Specification” (version 2.00), and the “SDIO Simplified Specification” (version 2.00). These documents are available for download by visiting the SD Association web site at: [http://www.sdcard.org/downloads/pls/simplified_specs/archive/index.html](http://www.sdcard.org/downloads/pls/simplified_specs/archive/index.html).
Register 57-5:  SDHCDATA: SDHC Data Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>23:16</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>15:8</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>7:0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
</tbody>
</table>

Legend:
- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as ‘0’
- -n = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

bit 31-0  **DATA<31:0>:** Buffer Data bits
These bits are used to access bits 31 through 0 of the internal data buffer.
## Section 57. Secure Digital Host Controller (SDHC)

### Register 57-6: SDHCSTAT1: SDHC Status Register 1

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>30:22/14/6</td>
<td>29:21/13/5</td>
<td>28/20/12/4</td>
<td>27/19/11/3</td>
<td>26/18/10/2</td>
<td>25/17/9/1</td>
<td>24/16/8/0</td>
<td></td>
</tr>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R-x, HC</td>
<td></td>
</tr>
<tr>
<td>23:16</td>
<td>R-x, HC</td>
<td>R-x, HC</td>
<td>R-x, HC</td>
<td>R-x, HC</td>
<td>R-x, HC</td>
<td>R-x, HC</td>
<td>R-x, HC</td>
<td></td>
</tr>
<tr>
<td>15:8</td>
<td>DATA3SLVL</td>
<td>DATA2SLVL</td>
<td>DATA1SLVL</td>
<td>DATA0SLVL</td>
<td>WPSLVL</td>
<td>CDSLVL</td>
<td>CARDST</td>
<td>CARDINS</td>
</tr>
<tr>
<td>7:0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R-0, HC</td>
<td>R-0, HC</td>
<td>R-0, HC</td>
<td>R-0, HC</td>
</tr>
</tbody>
</table>

**Legend:**
- **HC** = Hardware Cleared
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as '0'

**Note:** This register is used to recover from errors and for debugging.
Register 57-6:   SDHCSTAT1: SDHC Status Register 1 (Continued)

bit 10  **BWEN**: Buffer Write Enable bit
       1 = Buffer write is enabled
       0 = Buffer write is disabled

bit 9   **RDACTIVE**: Read Transfer Active bit
       1 = Data is being transferred
       0 = No valid data

bit 8   **WRACTIVE**: Write Transfer Active bit
       1 = Data is being transferred
       0 = No valid data

bit 7-3 **Unimplemented**: Read as ‘0’

bit 2   **DLACTIVE**: DAT Line Active bit
       1 = DAT line is active
       0 = DAT line is inactive

bit 1   **CINHDAT**: Command Inhibit (DAT) bit
       1 = A command that uses the DAT line cannot be issued
       0 = A command that uses the DAT line can be issued

bit 0   **CINHCMDS**: Command Inhibit (CMD) bit
       1 = A command cannot be issued
       0 = A command can only be issued using the CMD line

**Note:** This register is used to recover from errors and for debugging.
## Register 57-7: SDHCCON1: SDHC Control Register 1

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
<th>Bit Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>31/23/15/7</td>
<td>23:16</td>
<td>23/16</td>
<td>15:8</td>
<td>15/8</td>
<td>7:0</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>22/14/6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>30/22/14/6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>29/21/13/5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>28/20/12/4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>27/19/11/3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>26/18/10/2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>25/17/9/1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>24/16/8/0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>Unimplemented: Read as ‘0’</td>
</tr>
<tr>
<td>23:16</td>
<td>U-0</td>
<td>Unimplemented: Read as ‘0’</td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>Unimplemented: Read as ‘0’</td>
</tr>
<tr>
<td>7:0</td>
<td>R/W-0</td>
<td>CDSSEL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CDTLVL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DMASEL&lt;1:0&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HSEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DTXWIDTH</td>
</tr>
</tbody>
</table>

### Legend:
- **HC** = Hardware Cleared
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- **x** = Bit is unknown

- **bit 31-27** Unimplemented: Read as ‘0’
- **bit 26** **WKONREM**: Wake-up Event Enable on SD Card Removal bit
  - 1 = Wake-up event is enabled
  - 0 = Wake-up event is disabled
- **bit 25** **WKONINS**: Wake-up Event Enable on SD Card Insertion bit
  - 1 = Wake-up event is enabled
  - 0 = Wake-up event is disabled
- **bit 24** **WKONINT**: Wake-up Event Enable on SD Card Interrupt bit
  - 1 = Wake-up event is enabled
  - 0 = Wake-up event is disabled
- **bit 23-20** Unimplemented: Read as ‘0’
- **bit 19** **INTBG**: Interrupt at Block Gap bit
  - 1 = Interrupt is enabled
  - 0 = Interrupt is disabled
- **bit 18** **RDWTCON**: Read Wait Control bit
  - 1 = Read wait control is enabled
  - 0 = Read wait control is disabled
- **bit 17** **CONTREQ**: Continue Request bit
  - A write to this bit is ignored if STOPREQ is set to ‘1’.
  - 1 = Restart
  - 0 = No effect
- **bit 16** **SBGREQ**: Stop at Block Gap Request bit
  - 1 = Stop
  - 0 = Transfer
- **bit 15-9** Unimplemented: Read as ‘0’
- **bit 8** **SDBP**: SD Bus Power bit
  - 1 = Bus power is on
  - 0 = Bus power is off
- **bit 7** **CDSEL**: Card Detect Signal Selection bit
  - 1 = The card detect test level is select (for test purposes)
  - 0 = SDCDx is selected (for normal use)
- **bit 6** **CDTLVL**: Card Detect Test Level bit
  - 1 = Card is inserted
  - 0 = Card is not inserted
- **bit 5** Unimplemented: Read as ‘0’
Register 57-7: SDHCCON1: SDHC Control Register 1 (Continued)

bit 4-3  DMASEL<1:0>: DMA Select bits
        11 = Reserved
        10 = 32-bit address ADMA2 is selected
        01 = Reserved
        00 = Reserved

bit 2  HSEN: High-Speed Enable bit
        1 = High-Speed mode is enabled
        0 = Normal Speed mode is enabled

bit 1  DTXWIDTH: Data Transfer Width bit
        1 = 4-bit mode
        0 = 1-bit mode

bit 0  Unimplemented: Read as '0'
### Register 57-8: SDHCON2: SDHC Control Register 2

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>23:16</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>15:8</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>7:0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
</tbody>
</table>

**Legend:**
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- *x* = Bit is unknown
- **HC** = Hardware Cleared

#### bit 31-27
**Unimplemented:** Read as ‘0’

#### bit 26
**SWRDATA:** Software Reset for DAT Line bit
- **1** = DMA and part of the data logic are reset
- **0** = Continue operation

#### bit 25
**SWRCMD:** Software Reset for CMD Line bit
- **1** = Clears Present State and Interrupt Status registers and CMD bits
- **0** = Continue operation

#### bit 24
**SWRALL:** Software Reset for All bit
- **1** = Issue reset command and reinitialize the SD card
- **0** = Divided Clock mode is selected

#### bit 23-20
**Unimplemented:** Read as ‘0’

#### bit 19-16
**DTOC<3:0>:** Data Time-out Counter Value bits
- 1111 = Time-out clock x 2^27
- 1110 = Time-out clock x 2^26
- ...
- 0000 = Time-out clock x 2^13

#### bit 15-8
**SDCLKDIV<7:0>:** SDCLK Divider Select bits
When 8-bit Divided Clock mode is selected:
- 0x80 - Base clock divided by 256
- 0x40 - Base clock divided by 128
- 0x20 - Base clock divided by 64
- 0x10 - Base clock divided by 32
- 0x08 - Base clock divided by 16
- 0x04 - Base clock divided by 8
- 0x02 - Base clock divided by 4
- 0x01 - Base clock divided by 2
- 0x00 - Base clock

#### bit 7-3
**Unimplemented:** Read as ‘0’

#### bit 2
**SDCLKEN:** SD Clock Enable bit
- **1** = SD clock is enabled
- **0** = SD clock is disabled

#### bit 1
**ICLKSTABLE:** Internal Clock Stable bit
- **1** = Internal clock is ready
- **0** = Internal clock is not ready

#### bit 0
**ICLKEN:** Internal Clock Enable bit
- **1** = Oscillate
- **0** = Stop
Register 57-9: SDHCINTSTAT: SDHC Interrupt Status Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 29/21/13/5</th>
<th>Bit 27/19/11/3</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td>31:24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td>23:16</td>
<td>CLEIF</td>
<td>DEBEIF</td>
<td>DCRCEIF</td>
<td>DTOEIF</td>
<td>CIDEIF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CEBEIF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CCRCEIF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CTOEIF</td>
</tr>
<tr>
<td>15:8</td>
<td>R-0, HC</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R-0, HC</td>
</tr>
<tr>
<td></td>
<td>EIF</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7:0</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td></td>
<td>CARDRIF</td>
<td>CARDIIF</td>
<td>BRRDYIF</td>
<td>BWRDYIF</td>
<td>DMAIF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>BGIF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TXCIF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CEIF</td>
</tr>
</tbody>
</table>

Legend:
HC = Hardware Cleared
R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’

-bit 31-26 Unimplemented: Read as ‘0’
-bit 25 ADEIF: ADMA Error Interrupt Flag bit
1 = ADMA error has occurred
0 = ADMA error has not occurred
-bit 24 ACEIF: Auto CMD12 Error Interrupt Flag bit
1 = Auto CMD12 error has occurred
0 = Auto CMD12 error has not occurred
-bit 23 CLEIF: Current-Limit Error Interrupt Flag bit
1 = Current-limit error has occurred
0 = Current-limit error has not occurred
-bit 22 DEBEIF: Data End Bit Error Interrupt Flag bit
1 = Data End bit error has occurred
0 = Data End bit error has not occurred
-bit 21 DCRCEIF: Data CRC Error Interrupt Flag bit
1 = Data CRC error has occurred
0 = Data CRC error has not occurred
-bit 20 DTOEIF: Data Time-out Error Interrupt Flag bit
1 = Data time-out error has occurred
0 = Data time-out error has not occurred
-bit 19 CIDEIF: Command Index Error Interrupt Flag bit
1 = Command index error has occurred
0 = Command index error has not occurred
-bit 18 CEBEIF: Command End Bit Error Interrupt Flag bit
1 = End bit error was generated
0 = End bit error was not generated
-bit 17 CCRCEIF: Command CRC Error Interrupt Flag bit
1 = Command CRC error has occurred
0 = Command CRC error has not occurred
-bit 16 CTOEIF: Command Time-out Error Interrupt Flag bit
1 = Command time-out error has occurred
0 = Command time-out error has not occurred
-bit 15 EIF: Error Interrupt Flag bit
This bit is set if any or all bits, 0 through 9, in this register are set.
1 = Error was detected
0 = No error was detected
Register 57-9: SDHCINTSTAT: SDHC Interrupt Status Register (Continued)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>14-9</td>
<td>Unimplemented</td>
<td>Read as '0'</td>
</tr>
<tr>
<td>8</td>
<td>CARDIF: Card Interrupt Status bit</td>
<td>1 = Generate card interrupt, 0 = Do not generate card interrupt</td>
</tr>
<tr>
<td>7</td>
<td>CARDRIF: Card Removal Interrupt Flag bit</td>
<td>1 = Card has been removed, 0 = Card state is stable or debouncing</td>
</tr>
<tr>
<td>6</td>
<td>CARDIIF: Card Insertion Interrupt Flag bit</td>
<td>1 = Card has been inserted, 0 = Card state is stable or debouncing</td>
</tr>
<tr>
<td>5</td>
<td>BRRDYIF: Buffer Read Ready Interrupt Flag bit</td>
<td>1 = Ready to read buffer, 0 = Not ready to read buffer</td>
</tr>
<tr>
<td>4</td>
<td>BWRDYIF: Buffer Write Ready Interrupt Flag bit</td>
<td>1 = Ready to write buffer, 0 = Not ready to write buffer</td>
</tr>
<tr>
<td>3</td>
<td>DMAIF: DMA Interrupt Status bit</td>
<td>1 = DMA interrupt was generated, 0 = DMA interrupt was not generated</td>
</tr>
<tr>
<td>2</td>
<td>BGIF: Block Gap Interrupt Flag bit</td>
<td>1 = Transaction stopped at block gap, 0 = No block gap event has occurred</td>
</tr>
<tr>
<td>1</td>
<td>TXCIF: Transfer Complete Interrupt Flag bit</td>
<td>1 = Command execution has completed, 0 = Command execution has not completed</td>
</tr>
<tr>
<td>0</td>
<td>CEIF: Command Complete Interrupt Flag bit</td>
<td>1 = Command is complete, 0 = Command is not complete</td>
</tr>
</tbody>
</table>
### Register 57-10: SDHCINTEN: SDHC Interrupt Flag Enable Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 28/20/12/4</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td>23:16</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td></td>
<td>CLEFIE</td>
<td>DEBEFIE</td>
<td>DCRCEFIE</td>
<td>DTOEFIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CIDXEFIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CDEBEFIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CCRCEFIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CTOEFIE</td>
</tr>
<tr>
<td>15:8</td>
<td>R-0, HC</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R-0, HC</td>
</tr>
<tr>
<td></td>
<td>FTZIE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CARDIE</td>
</tr>
<tr>
<td></td>
<td>CARDRIE</td>
<td>CARDIIE</td>
<td>BRRDYIE</td>
<td>BWRDYIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DMAIE</td>
<td>BGIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TXEIE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CEIE</td>
</tr>
</tbody>
</table>

**Legend:**
- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as '0'
- -n = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

**Legend:**
- HC = Hardware Cleared

**bit 31-26** Unimplemented: Read as ‘0’

**bit 25**
- **ADEFIE**: ADMA Interrupt Flag Error Enable bit
  - 1 = ADMA error interrupt flag is enabled
  - 0 = ADMA error interrupt flag is masked

**bit 24**
- **ACEFIE**: Auto CMD12 Interrupt Flag Error Enable bit
  - 1 = Auto CMD12 error interrupt flag is enabled
  - 0 = Auto CMD12 error interrupt flag is masked

**bit 23**
- **CLEFIE**: Current-Limit Interrupt Flag Error Enable bit
  - 1 = Current-limit error interrupt flag is enabled
  - 0 = Current-limit error interrupt flag is masked

**bit 22**
- **DEBEFIE**: Data End Bit Interrupt Flag Error Enable bit
  - 1 = Data End bit error interrupt flag is enabled
  - 0 = Data End error interrupt flag is masked

**bit 21**
- **DCRCEFIE**: Data CRC Interrupt Flag Error Enable bit
  - 1 = Data CRC error interrupt flag is enabled
  - 0 = Data CRC error interrupt flag is masked

**bit 20**
- **DTOEFIE**: Data Time-out Interrupt Flag Error Enable bit
  - 1 = Data time-out interrupt flag is enabled
  - 0 = Data time-out interrupt flag is masked

**bit 19**
- **CIDXEFIE**: Command Index Interrupt Flag Error Enable bit
  - 1 = Command index error interrupt flag is enabled
  - 0 = Command index error interrupt flag is masked

**bit 18**
- **CDEBEFIE**: Command End Bit Interrupt Flag Error Enable bit
  - 1 = Command End bit error interrupt flag is enabled
  - 0 = Command End bit error interrupt flag is masked

**bit 17**
- **CCRCEFIE**: Command CRC Interrupt Flag Error Enable bit
  - 1 = Command CRC error interrupt flag is enabled
  - 0 = Command CRC error interrupt flag is masked

**bit 16**
- **CTOEFIE**: Command Time-out Interrupt Flag Error Enable bit
  - 1 = Command time-out interrupt flag is enabled
  - 0 = Command time-out interrupt flag is masked

**bit 15**
- **FTZIE**: Fixed to Zero Interrupt Flag Enable bit
  - This bit is set if any or all bits, 0 through 9, in this register are set.
  - 1 = Error was detected
  - 0 = No error was detected
Register 57-10:  SDHCINTEN: SDHC Interrupt Flag Enable Register (Continued)

bit 14-9  Unimplemented: Read as '0'

bit 8  CARDIE: Card Interrupt Flag Enable bit
1 = Card interrupt flag is enabled
0 = Card interrupt flag is masked

bit 7  CARDRIE: Card Removal Interrupt Flag Enable bit
1 = Card removal interrupt flag is enabled
0 = Card removal interrupt flag is masked

bit 6  CARDIIE: Card Insertion Interrupt Flag Enable bit
1 = Card insertion interrupt flag is enabled
0 = Card insertion interrupt flag is masked

bit 5  BRRDYIE: Buffer Read Ready Interrupt Flag Enable bit
1 = Buffer read ready interrupt flag is enabled
0 = Buffer read ready interrupt flag is masked

bit 4  BWRDYIE: Buffer Write Ready Interrupt Flag Enable bit
1 = Buffer write ready interrupt flag is enabled
0 = Buffer write ready interrupt flag is masked

bit 3  DMAIE: DMA Interrupt Flag Enable bit
1 = DMA interrupt flag is enabled
0 = DMA interrupt flag is masked

bit 2  BGIE: Block Gap Interrupt Flag Enable bit
1 = Block gap event interrupt flag is enabled
0 = Block gap event interrupt flag is masked

bit 1  TXEIE: Transfer Complete Interrupt Flag Enable bit
1 = Transfer complete interrupt flag is enabled
0 = Transfer complete interrupt flag is masked

bit 0  CEIE: Command Complete Interrupt Flag Enable bit
1 = Command complete interrupt flag is enabled
0 = Command complete interrupt flag is masked
### Register 57-11: SDHCINTSEN: SDHC Interrupt Signal Enable Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R/W-0, HC</td>
</tr>
<tr>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>ADEISE</td>
</tr>
<tr>
<td>bit 31-26</td>
<td>Unimplemented: Read as ‘0’</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 25</td>
<td>ADEISE: ADMA Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = ADMA error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = ADMA error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 24</td>
<td>ACEISE: Auto CMD12 Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Auto CMD12 error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Auto CMD12 error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 23</td>
<td>CLEISE: Current-Limit Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Current-limit error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Current-limit error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 22</td>
<td>DEBEISE: Data End Bit Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Data end bit error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Data end bit error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 21</td>
<td>DCRCEISE: Data CRC Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Data CRC error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Data CRC error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 20</td>
<td>DTOEISE: Data Time-out Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Data time-out error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Data time-out error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 19</td>
<td>CIDXEISE: Command Index Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Command index error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Command index error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 18</td>
<td>CEBEISE: Command End Bit Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Command End bit error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Command End bit error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 17</td>
<td>CCRCEISE: Command CRC Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Command CRC error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Command CRC error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 16</td>
<td>CTOEISE: Command Time-out Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Command time-out error signal is enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Command time-out error signal is masked</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 15</td>
<td>FTZEISE: Fixed to Zero Error Interrupt Signal Enable bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>This bit is set if any or all bits, 0 through 9, in this register are set.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Error was detected</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = No error was detected</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as ‘0’
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

HC = Hardware Cleared
### Register 57-11: SDHCINTSEN: SDHC Interrupt Signal Enable Register (Continued)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>14-9</td>
<td>Unimplemented</td>
<td>Read as '0'</td>
</tr>
<tr>
<td>8</td>
<td><strong>CARDISE</strong>: Card Interrupt Signal Enable bit</td>
<td>1 = Card interrupt signal is enabled, 0 = Card interrupt signal is masked</td>
</tr>
<tr>
<td>7</td>
<td><strong>CARDRISE</strong>: Card Removal Interrupt Signal Enable bit</td>
<td>1 = Card removal signal is enabled, 0 = Card removal signal is masked</td>
</tr>
<tr>
<td>6</td>
<td><strong>CARDIISE</strong>: Card Insertion Interrupt Signal Enable bit</td>
<td>1 = Card insertion signal is enabled, 0 = Card insertion signal is masked</td>
</tr>
<tr>
<td>5</td>
<td><strong>BRRDYISE</strong>: Buffer Read Ready Interrupt Signal Enable bit</td>
<td>1 = Buffer read ready signal is enabled, 0 = Buffer read ready signal is masked</td>
</tr>
<tr>
<td>4</td>
<td><strong>BWRDYISE</strong>: Buffer Write Ready Interrupt Signal Enable bit</td>
<td>1 = Buffer write ready signal is enabled, 0 = Buffer write ready signal is masked</td>
</tr>
<tr>
<td>3</td>
<td><strong>DMAISE</strong>: DMA Interrupt Signal Enable bit</td>
<td>1 = DMA interrupt signal is enabled, 0 = DMA interrupt signal is masked</td>
</tr>
<tr>
<td>2</td>
<td><strong>BGISE</strong>: Block Gap Interrupt Signal Enable bit</td>
<td>1 = Block gap event signal is enabled, 0 = Block gap event signal is masked</td>
</tr>
<tr>
<td>1</td>
<td><strong>TXEISE</strong>: Transfer Complete Interrupt Signal Enable bit</td>
<td>1 = Transfer complete signal is enabled, 0 = Transfer complete signal is masked</td>
</tr>
<tr>
<td>0</td>
<td><strong>CEISE</strong>: Command Complete Interrupt Signal Enable bit</td>
<td>1 = Command complete signal is enabled, 0 = Command complete signal is masked</td>
</tr>
</tbody>
</table>
Register 57-12: SDHCSTAT2: SDHC Status Register 2

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>23:16</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>7:0</td>
<td>CNISSE</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

Legend:

HC = Hardware Cleared
R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’

 bit 31-8 **Unimplemented:** Read as ‘0’

 bit 7 **CNISSE:** Command Not Issued by Auto CMD12 Error bit
1 = Command was not issued
0 = No error

 bit 6-5 **Unimplemented:** Read as ‘0’

 bit 4 **ACIDX:** Auto CMD12 Index Error bit
1 = Index error was generated
0 = Index error was not generated

 bit 3 **ACEBE:** Auto CMD12 End Bit Error bit
1 = End bit error was generated
0 = End bit error was not generated

 bit 2 **ACCRCE:** Auto CMD12 CRC Error bit
1 = CRC error was generated
0 = CRC error was not generated

 bit 1 **ACTOE:** Auto CMD12 Time-out Error bit
1 = Time-out error was generated
0 = Time-out error was not generated

 bit 0 **ACNEXEC:** Auto CMD12 Not Executed bit
1 = Auto CMD12 was not executed
0 = Auto CMD12 was executed
### Register 57-13: SDHCCAP: SDHC Capabilities Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R-1, HS</td>
</tr>
<tr>
<td>23:16</td>
<td>R-x, HS</td>
<td>U-0</td>
<td>R-x, HS</td>
<td>U-0</td>
<td>U-0</td>
<td>R-0, HS</td>
<td>R-0, HS</td>
<td></td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>U-0</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
</tr>
<tr>
<td>7:0</td>
<td>R-0</td>
<td>U-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
</tr>
</tbody>
</table>

**Legend:**
- HS = Hardware settable
- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as ‘0’
- -n = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

- **bit 31-25 Unimplemented:** Read as ‘0’
- **bit 24 3V3:** 3.3V Voltage Support bit
  - 1 = Voltage of 3.3V is supported
- **bit 23 SRESUME:** Suspend/Resume Support bit
  - 1 = Suspend/resume is supported
  - 0 = Suspend/resume is not supported
- **bit 22 Unimplemented:** Read as ‘0’
- **bit 21 HISPEED:** High-speed Support bit
  - 1 = High speed is supported
  - 0 = High speed is not supported
- **bit 20 Unimplemented:** Read as ‘0’
- **bit 19 ADMA2:** ADMA2 Support bit
  - 1 = ADMA2 is supported
  - 0 = ADMA2 is not supported
- **bit 18 Unimplemented:** Read as ‘0’
- **bit 17-16 MBLEN<1:0>:** Maximum Block Length bits
  - 11 = Reserved
  - 10 = 2048
  - 01 = 1024
  - 00 = 512
- **bit 15-14 Unimplemented:** Read as ‘0’
- **bit 13-8 BASECLK<5:0>:** Base Clock Frequency for SDCLK bits
  - 111111 = 63 MHz
  - 111110 = 62 MHz
  - 111101 = 61 MHz
  - ...
  - 000010 = 2 MHz
  - 000001 = 1 MHz
  - 000000 = Reserved
- **bit 7 TOCLKU:** Time-out Clock Unit bit
  - 1 = Time-out clock unit is in kHz
  - 0 = Time-out clock unit is in MHz
- **bit 6 Unimplemented:** Read as ‘0’
Register 57-13:  SDHCCAP: SDHC Capabilities Register (Continued)

bit 5-0  TOCLKFREQ<5:0>: Time-out Clock Frequency bits

The TOCLKU bit defines the unit, either kHz or MHz, of these bit values.

- 111111 = 63 kHz or 63 MHz
- 111110 = 62 kHz or 62 MHz
- 111101 = 61 kHz or 61 MHz
- 000010 = 2 kHz or 2 MHz
- 000001 = 1 kHz or 1 MHz
- 000000 = Reserved
## Register 57-14: SDHCMAXCAP: SDHC Maximum Current Capabilities Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>23:16</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>7:0</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
<td>R-x, HS</td>
</tr>
</tbody>
</table>

Legend:
- **HS** = Hardware Set
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- ‘-n’ = Value at POR
  - ‘1’ = Bit is set
  - ‘0’ = Bit is cleared
  - **x** = Bit is unknown

**bit 31-8**  **Unimplemented**: Read as ‘0’

**bit 7-0**  **MC3V3<7:0>:** Maximum Current for 3.3V bits

- 11111111 = Reserved
- 01111111 = Reserved
- 000000101 = Reserved
- 00000100 = 16 mA
- 00000011 = 12 mA
- 00000010 = 8 mA
- 00000001 = 4 mA
- 00000000 = Reserved
Register 57-15: SDHCFE: SDHC Force Event Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/24</th>
<th>Bit 23/16</th>
<th>Bit 15/8</th>
<th>Bit 7:0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>W-0, HC</td>
<td>U-0</td>
<td>W-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
<td>W-0, HC</td>
</tr>
<tr>
<td>23:16</td>
<td>FEACLE</td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
</tr>
<tr>
<td></td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
<td>FEIDXE</td>
</tr>
<tr>
<td></td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
<td>FECEBE</td>
</tr>
<tr>
<td></td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
<td>FECEBE</td>
</tr>
<tr>
<td></td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
<td>FECEBE</td>
</tr>
<tr>
<td></td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
<td>FECEBE</td>
</tr>
<tr>
<td></td>
<td>FEDEBE</td>
<td>FEDCRCE</td>
<td>FEDTOE</td>
<td>FECEBE</td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td></td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>7:0</td>
<td>FECNIACE</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>FECNIACE</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>FECNIACE</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>FECNIACE</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

Legend:
- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as ‘0’
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

bit 31-26: Unimplemented: Read as ‘0’

bit 25: **FEADE**: Force Event for ADMA Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 24: **FEACE**: Force Event for Auto CMD 12 Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 23: **FECLE**: Force Event for Current-Limit Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 22: **FEDEBE**: Force Event for Data End Bit Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 21: **FEDCRCE**: Force Event for Data CRC Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 20: **FEDTOE**: Force Event for Data Time-out Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 19: **FEIDXE**: Force Event for Command Index Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 18: **FECEBE**: Force Event for Command End Bit Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 17: **FECCRCE**: Force Event for Command CRC Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 16: **FECTOE**: Force Event for Command Time-out Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated

bit 15-8: Unimplemented: Read as ‘0’

bit 7: **FECNIACE**: Force Event for Command Not Issued by Auto CMD12 Error bit
- 1 = Interrupt was generated
- 0 = Interrupt was not generated
Register 57-15: SDHCFE: SDHC Force Event Register (Continued)

bit 6-5  **Unimplemented**: Read as '0'

bit 4  **FEACIDX**: Force Event for Auto CMD12 Index Error bit
   1 = Interrupt was generated
   0 = Interrupt was not generated

bit 3  **FEACEBE**: Force Event for Auto CMD12 End Bit Error bit
   1 = Interrupt was generated
   0 = Interrupt was not generated

bit 2  **FEACCRCE**: Force Event for Auto CMD12 CRC Error bit

bit 1  **FEACTOE**: Force Event for Auto CMD12 Time-out Error bit
   1 = Interrupt was generated
   0 = Interrupt was not generated

bit 0  **FEACNEE**: Force Event for Auto CMD12 Not Executed Error bit
   1 = Interrupt was generated
   0 = Interrupt was not generated
Register 57-16: SDHCADESTAT: SDHC ADMA Error Status Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>23:16</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>15:8</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
</tr>
<tr>
<td>7:0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>U-0</td>
<td>R-0, HC</td>
<td>R-0, HC</td>
<td>R-0, HC</td>
</tr>
</tbody>
</table>

Legend:
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- **x** = Bit is unknown

bit 31-3 Unimplemented: Read as ‘0’

bit 2 **ADLMERR**: ADMA Length Mismatch Error bit
- 1 = Length mismatch error has occurred
- 0 = Length mismatch error has not occurred

bit 1-0 **ADERRST<1:0>**: ADMA Error State bits
- 11 = Data transfer error
- 10 = Reserved
- 01 = Fetch descriptor error
- 00 = Stop DMA error

Register 57-17: SDHCAADDR: SDHC ADMA Address Register

<table>
<thead>
<tr>
<th>Bit Range</th>
<th>Bit 31/23/15/7</th>
<th>Bit 30/22/14/6</th>
<th>Bit 29/21/13/5</th>
<th>Bit 28/20/12/4</th>
<th>Bit 27/19/11/3</th>
<th>Bit 26/18/10/2</th>
<th>Bit 25/17/9/1</th>
<th>Bit 24/16/8/0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31:24</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>23:16</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>15:8</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
<tr>
<td>7:0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
<td>R/W-0</td>
</tr>
</tbody>
</table>

Legend:
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- **x** = Bit is unknown

bit 31-0 **ADDR<31:0>**: ADMA Address Register bits
- These bits contain the address of the executing command of the ADMA descriptor table.
57.3 SDHC MODULE OPERATION

For information on SDHC operation, refer to the following specifications:

- “SD Host Controller Simplified Specification” (version 2.00)
- “Physical Layer Simplified Specification” (version 2.00)
- “SDIO Simplified Specification” (version 2.00)

These documents are available for download by visiting the SD Association web site at: http://www.sdcard.org/downloads/pls/simplified_specs/archive/index.html
57.4 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the PIC32 device family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Secure Digital Host Controller (SDHC) are:

<table>
<thead>
<tr>
<th>Title</th>
<th>Application Note #</th>
</tr>
</thead>
<tbody>
<tr>
<td>No related application notes at this time.</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Note:** Please visit the Microchip web site ([www.microchip.com](http://www.microchip.com)) for additional application notes and code examples for the PIC32 family of devices.
57.5 REVISION HISTORY

Revision A (June 2015)

This is the initial released version of this document.
Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, flexPWR, JukeBox, KEELOGO, KEELOGO logo, Kleer, LANCheck, MediaLB, MOST, MOST logo, MPLAB, OptoLyzer, PIC, PICSTART, PIC12 logo, RightTouch, SpyNIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

The Embedded Control Solutions Company and mTouch are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, ECAN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, KleerNet, KleerNet logo, MiWi, MPASM, MFP, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, RightTouch logo, REAL ICE, SQI, Serial Quad I/O, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademarks of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2015, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.
