Date: Oct. 01, 2013

RENESAS TECHNICAL UPDATE

1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan
Renesas Electronics Corporation

Product Category	MPU/MCU		Document No.	TN-SH7-A875A/E	Rev.	1.00
Title	SH7455 Group, SH7456 Group User's Manual Hardware Errata Rev. B		Information Category	Technical Notification		
Applicable Product	SH7455 Group, SH7456 Group	Lot No.	Reference Document	SH7455 Group, SH7456 Group User's Manual: Hardware Rev. 1.10 (R01UH0030EJ0110)		

We inform you of the corrections of "SH7455 Group, SH7456 Group User's Manual: Hardware Rev. 1.10 (Published on September 22, 2011)".

When you use "SH7455 Group, SH7456 Group User's Manual: Hardware Rev. 1.10", should be used together the attached errata.

In addition, the corrections in the following are also included in the attached errata (Rev. B).

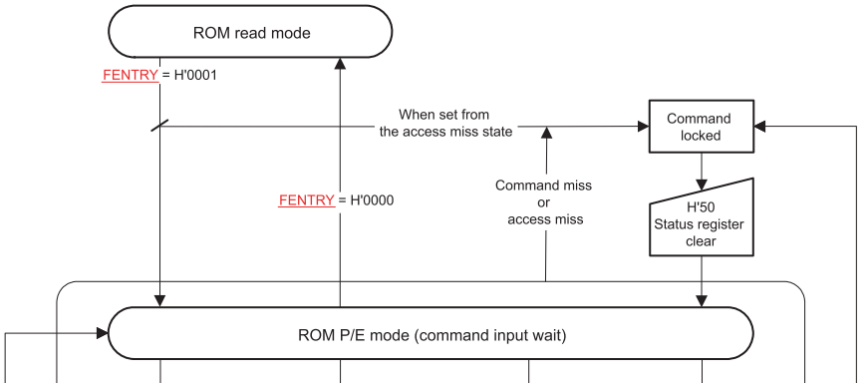
- Technical update TN-SH7-A827A/E: Errata (Rev. A)
- Technical update TN-SH7-A859A/E: Errata to User's Manual Regarding CAN Module

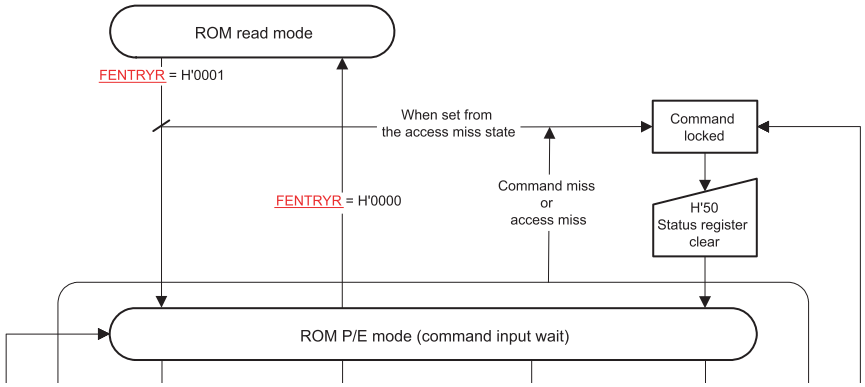
Attached Document: "SH7455 Group, SH7456 Group User's Manual: Hardware Rev. 1.10" Errata Rev. B – 11 sheets

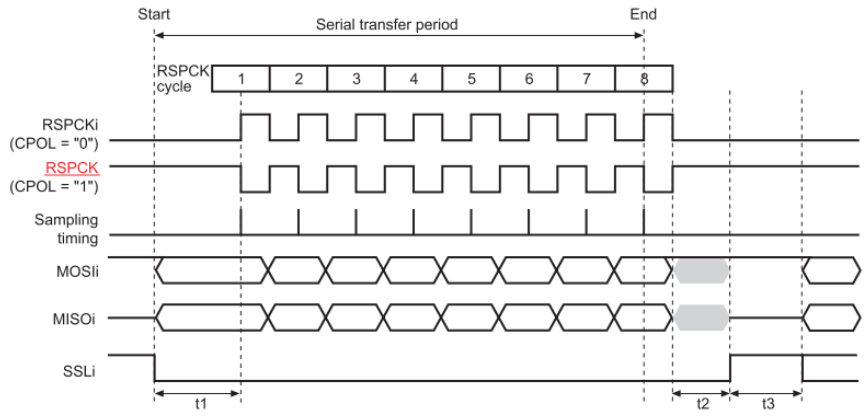
*Changes/additions are written in reds and underlined.

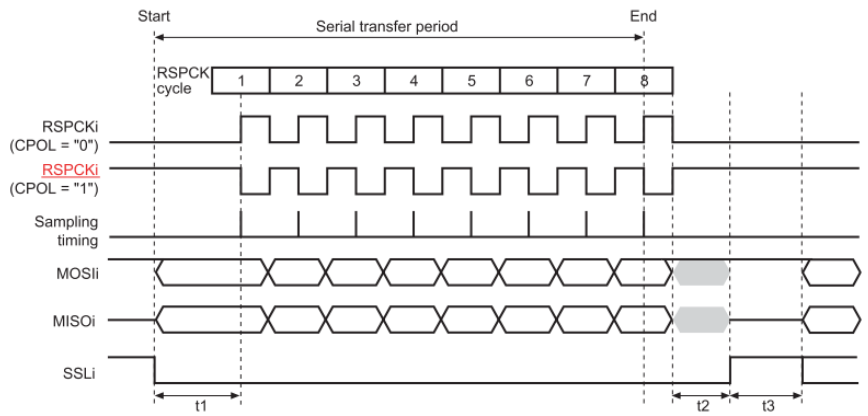
<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. A</td><td>25-23</td><td>25.4.6 (3) Receive Operation</td><td>Incorrect description is corrected (the 12th line).
Error: 4. To stop receiving when MST bit = "1xx", set RCVD bit in the ICCR1 register to "1", then read the ICDRR register.
Correction: 4. To stop receiving when MST bit = "1", set RCVD bit in the ICCR1 register to "1", then read the ICDRR register.</td></tr>
<tr><td>Added in Rev. A</td><td>32-76</td><td>32.7.1 FlexRay CC Status Vector Register</td><td>Description of the bit 29 to 24 (PSL5 to PSL0 bit) in the FlexRay CC Status Vector Register (FRCCSV) is corrected.
Error: Set to B'000100 when leaving HALT state.
Correction: Set to B'000000 when leaving HALT state.</td></tr>
<tr><td>Added in Rev. A</td><td>38-6</td><td>Table 38.6 DC Characteristics - Output Level Voltage: When 3.3 V is Used with Driving Ability Set to "Increased"</td><td>Table 38.6 DC Characteristics - Output Level Voltage: When 3.3 V is Used with Driving Ability Set to "Increased" : Incorrect description is corrected.
Error:
<table>
<tr><th>Item</th><th></th><th>Symbol</th><th>Min.</th><th>Unit</th></tr>
<tr><td>Output high-level voltage (normal output and driving ability)*1</td><td>PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9</td><td>V_{OH}</td><td>Vcc –1.1</td><td>V</td></tr>
</table>
Correction:
<table>
<tr><th>Item</th><th></th><th>Symbol</th><th>Min.</th><th>Unit</th></tr>
<tr><td>Output high-level voltage (normal output and driving ability)*</td><td>PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9</td><td>V_{OH}</td><td>Vcc –0.5</td><td>V</td></tr>
</table>
Error:
<table>
<tr><th>Item</th><th></th><th>Symbol</th><th>Max.</th><th>Unit</th></tr>
<tr><td>Output low-level voltage (normal output and driving ability)*1</td><td>PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9</td><td>V_{OL}</td><td>0.9</td><td>V</td></tr>
</table>
Correction:
<table>
<tr><th>Item</th><th></th><th>Symbol</th><th>Max.</th><th>Unit</th></tr>
<tr><td>Output low-level voltage (normal output and driving ability)*1</td><td>PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9</td><td>V_{OL}</td><td>0.4</td><td>V</td></tr>
</table></td></tr>
</table>

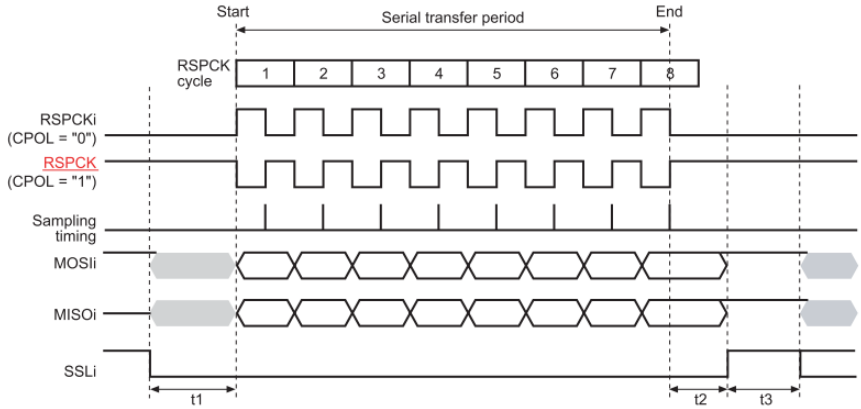
<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. A</td><td>38-22</td><td>Table 38.25 RSPI Timing</td><td>Table 38.25 RSPI Timing : Incorrect description is corrected.
Error:
<table>
<tr><th>Item</th><th></th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>Data input setup time</td><td>Slave</td><td>t_{SU}</td><td>25 + 2 x t_{cyc}</td><td>-</td><td>ns</td><td>38.20 to 38.23</td></tr>
</table>
Correction:
<table>
<tr><th>Item</th><th></th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>Data input setup time</td><td>Slave</td><td>t_{SU}</td><td>25 - 2 x t_{cyc}</td><td>-</td><td>ns</td><td>38.20 to 38.23</td></tr>
</table></td></tr>
<tr><td>Added in Rev. A</td><td>38-27</td><td>Table 38.28 DRI Timing (When Special Mode is On)</td><td>Table 38.28 DRI Timing (When Special Mode is On) : Incorrect description is corrected.
Error:
<table>
<tr><th>Item</th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>DIN3, DIN4 sampling edge undefined time before DIN1 initialization level release (when direct reset is selected)</td><td>tar</td><td>8</td><td>-</td><td>ns</td><td>38.25 to 38.28</td></tr>
<tr><td>DIN3, DIN4 sampling edge undefined time before DIN1 initialization level release</td><td>tbr</td><td>12</td><td>-</td><td>ns</td><td></td></tr>
</table>
Correction:
<table>
<tr><th>Item</th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>DIN3, DIN4 sampling edge undefined time before DIN1 initialization level release</td><td>tar</td><td>8</td><td>-</td><td>ns</td><td>38.25 to 38.28</td></tr>
<tr><td>DIN3, DIN4 sampling edge undefined time after DIN1 initialization level release</td><td>tbr</td><td>12</td><td>-</td><td>ns</td><td></td></tr>
</table></td></tr>
<tr><td>Added in Rev. A</td><td>38-28</td><td>Figure 38.28 Minimum Edge Count at DIN1 Initialization Level in Delayed Reset Mode</td><td>Figure 38.28 Minimum Edge Count at DIN1 Initialization Level in Delayed Reset Mode (Minimum Width at Initialization Level) : Incorrect description is corrected.
Error: DINn (n = 2, 3, 4)
Correction: DINn (n = 3, 4)</td></tr>
<tr><td>Added in Rev. A</td><td>38-34</td><td>Table 38.34 AUDR Module Timing (Vcc = 5.0 V)</td><td>Table 38.34 AUDR Module Timing (Vcc = 5.0 V) : Incorrect description is corrected.
Error:
<table>
<tr><th>Item</th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>AUDRD output delay time before AUDRCLK</td><td>td(AUDRCLKH-AUDRD)</td><td>-</td><td>35</td><td>ns</td><td>38.38</td></tr>
</table>
Correction:
<table>
<tr><th>Item</th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>AUDRD output delay time after AUDRCLK</td><td>td(AUDRCLKH-AUDRD)</td><td>-</td><td>35</td><td>ns</td><td>38.38</td></tr>
</table></td></tr>
<tr><td>Added in Rev. A</td><td>38-35</td><td>Table 38.35 AUDR Module Timing (Vcc = 3.3 V)</td><td>Table 38.35 AUDR Module Timing (Vcc = 3.3V) : Incorrect description is corrected.
Error:
<table>
<tr><th>Item</th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>AUDRD output delay time before AUDRCLK</td><td>td(AUDRCLKH-AUDRD)</td><td>-</td><td>40</td><td>ns</td><td>38.38</td></tr>
</table>
Correction:
<table>
<tr><th>Item</th><th>Symbol</th><th>Min.</th><th>Max.</th><th>Unit</th><th>Figures</th></tr>
<tr><td>AUDRD output delay time after AUDRCLK</td><td>td(AUDRCLKH-AUDRD)</td><td>-</td><td>40</td><td>ns</td><td>38.38</td></tr>
</table></td></tr>
</table>

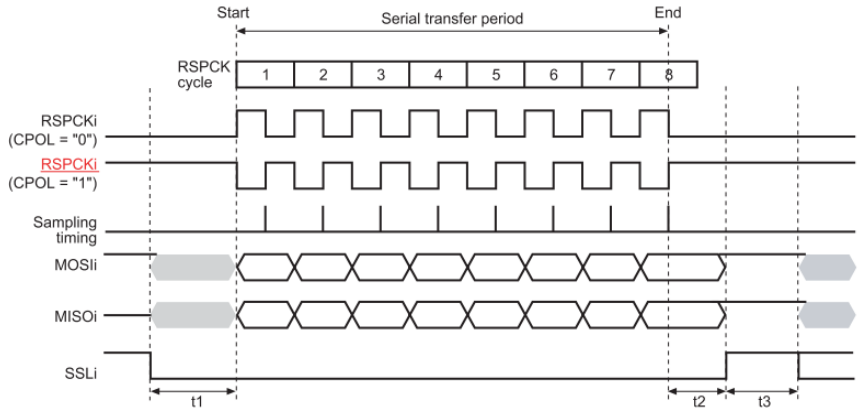
<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. B</td><td>12-7</td><td>12.3.2 Flash Access Status Register (FASTAT)</td><td>Description of the bit 7 (ROMAE bit) in the Flash Access Status Register (FASTAT) is corrected.
Error: An access command is issued to an address other than ROM program/erase addresses H'FD80 0000 to H'FD8F FFFF when the user boot MAT is selected.
Correction: An access command is issued to an address other than ROM program/erase addresses H'FD80 0000 to H'FD80 7FFF when the user boot MAT is selected.</td></tr>
<tr><td>Added in Rev. B</td><td>12-22</td><td>Figure 12.7 Command State Transitions in ROM Read Mode and P/E Mode</td><td>Figure 12.7 Command State Transitions in ROM Read Mode and P/E Mode : Incorrect description is corrected.
Error:
 

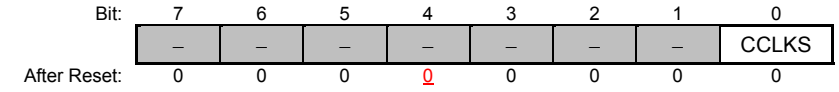
Correction:
 
</td></tr>
<tr><td>Added in Rev. B</td><td>12-35</td><td>12.9.4 Reset during Programming or Erasure</td><td>Description of Reset during Programming or Erasure is added.
-Description:
When a hardware reset by "L" level input to the RESET# pin, switching the power off, or a FCU reset by setting the FRESET bit in the FRESETR register, is executed during programming or erasure, the whole data in the programming or erasure area becomes undefined. When the data in an area have become undefined, erase the area before using it again.</td></tr>
</table>

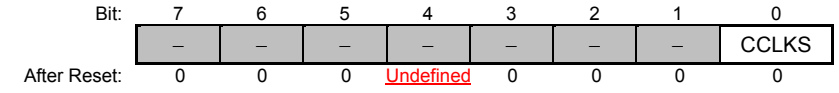
<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. B</td><td>24-10</td><td>24.3.3 RSPIi Pin Control Register (SPiPCR)</td><td>Description of the bit 5 (MOIFE bit) in the RSPIi Pin Control Register (SPiPCR) is corrected.
Error: - When the MOIFE bit is cleared to "0", RSPIi outputs on the MOSIi pin the last data unit from the previous serial transfer during the SSL negation period.
- 0: MOSIi output value equals final data from previous transfer
Correction: - When the MOIFE bit is cleared to "0", RSPIi outputs the final output level of the previous serial transfer to the MOSIi pin during the SSL negation period (When the CPHA bit is "0", MOSIi output value is undefined).
- 0: MOSIi output value equals final output level from previous transfer (When the CPHA bit is "0", MOSIi output value is undefined)</td></tr>
<tr><td>Added in Rev. B</td><td>24-25</td><td>24.3.13 RSPIi Command Registers 0 to 3 (SPiCMD0 to SPiCMD3)</td><td>Description of the bit 13 (SPNDEN bit) in the RSPIi Command Registers 0 to 3 (SPiCMD0 to SPiCMD3) is corrected.
Error: - If the SPNDEN bit is "0", the RSPIi sets the next-access delay to 1 RSPCK.
- 0: A next-access delay of 1 RSPCK
Correction: - If the SPNDEN bit is "0", the RSPIi sets the next-access delay to 1 RSPCK + 2 Pck.
- 0: A next-access delay of 1 RSPCK + 2 Pck</td></tr>
<tr><td>Added in Rev. B</td><td>24-29</td><td>Table 24.7 MOSIi Signal Value Determination during SSL Negation Period</td><td>Table 24.7 MOSIi Signal Value Determination during SSL Negation Period : Incorrect description is corrected.
Error:
<table>
<tr><th>MOIFE</th><th>MOIFV</th><th>MOSIi Signal Value during SSL Negation Period</th></tr>
<tr><td>0</td><td>0, 1</td><td>Final data from previous transfer</td></tr>
<tr><td>1</td><td>0</td><td>Always "L"</td></tr>
<tr><td>1</td><td>1</td><td>Always "H"</td></tr>
</table>
Correction:
<table>
<tr><th>MOIFE</th><th>MOIFV</th><th>MOSIi Signal Value during SSL Negation Period</th></tr>
<tr><td>0</td><td>0, 1</td><td>Final output level of the previous transfer (When the CPHA bit is "0", MOSIi output value is undefined)</td></tr>
<tr><td>1</td><td>0</td><td>Always "L"</td></tr>
<tr><td>1</td><td>1</td><td>Always "H"</td></tr>
</table></td></tr>
<tr><td>Added in Rev. B</td><td>24-35</td><td>Figure 24.11 RSPI Transfer Format (CPHA = "0")</td><td>Figure 24.11 RSPI Transfer Format (CPHA = "0") : Incorrect description is corrected.
Error:
 

Correction:
 
</td></tr>
</table>

<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. B</td><td>24-35</td><td>Figure 24.12 RSPI Transfer Format (CPHA = "1")</td><td>Figure 24.12 RSPI Transfer Format (CPHA = "1") : Incorrect description is corrected.
Error:
 

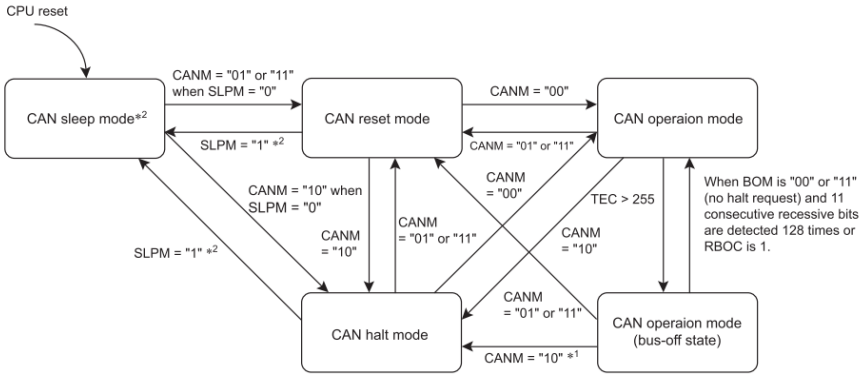
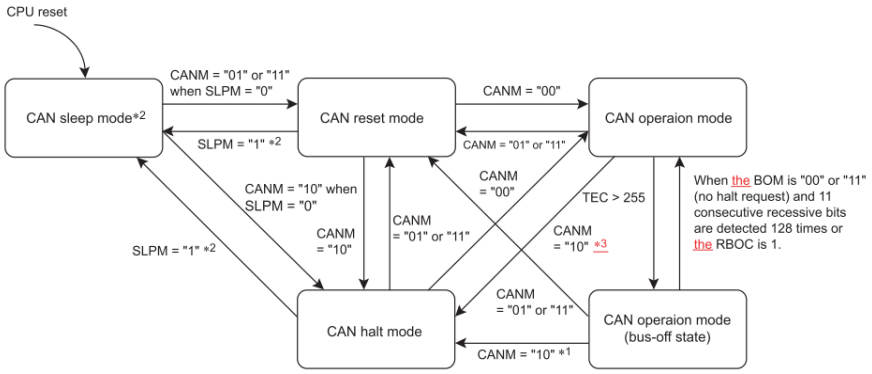
Correction:
 
</td></tr>
<tr><td>Added in Rev. B</td><td>26-5</td><td>Table 26.3 Register Configuration</td><td>The values after reset of the CANi Clock Select Register (CiCLKR) (i = 0 to 3) are corrected.
Error:
<table>
<tr><th>Register Name</th><th>Abbreviation</th><th>After Reset</th><th>P4 Address</th><th>Size</th><th>Page</th></tr>
<tr><td>CAN0 Clock Select Register</td><td>C0CLKR</td><td>H'00</td><td>H'FFFF 6847</td><td>8, 16, 32</td><td>26-15</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>CAN1 Clock Select Register</td><td>C1CLKR</td><td>H'00</td><td>H'FFFF 7847</td><td>8, 16, 32</td><td>26-15</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>CAN2 Clock Select Register</td><td>C2CLKR</td><td>H'00</td><td>H'FFFF 8847</td><td>8, 16, 32</td><td>26-15</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>CAN3 Clock Select Register</td><td>C3CLKR</td><td>H'00</td><td>H'FFFF 9847</td><td>8, 16, 32</td><td>26-15</td></tr>
</table>
Correction:
<table>
<tr><th>Register Name</th><th>Abbreviation</th><th>After Reset</th><th>P4 Address</th><th>Size</th><th>Page</th></tr>
<tr><td>CAN0 Clock Select Register</td><td>C0CLKR</td><td>Undefined</td><td>H'FFFF 6847</td><td>8, 16, 32</td><td>26-15</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>CAN1 Clock Select Register</td><td>C1CLKR</td><td>Undefined</td><td>H'FFFF 7847</td><td>8, 16, 32</td><td>26-15</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>CAN2 Clock Select Register</td><td>C2CLKR</td><td>Undefined</td><td>H'FFFF 8847</td><td>8, 16, 32</td><td>26-15</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>CAN3 Clock Select Register</td><td>C3CLKR</td><td>Undefined</td><td>H'FFFF 9847</td><td>8, 16, 32</td><td>26-15</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. B</td><td>26-15</td><td>26.3.2 CANi Clock Select Register (CiCLKR) (i = 0 to 3)</td><td>
The value after reset of the bit 4 in the CANi Clock Select Register (CiCLKR) (i = 0 to 3) is corrected.
Error:


<After Reset: H'00>
<table>
<tr><th>Bit</th><th>Abbreviation</th><th>After Reset</th><th>R</th><th>W</th><th>Description</th></tr>
<tr><td>4</td><td>–</td><td>0</td><td>?</td><td>0</td><td>Reserved Bit
Should be written with "0" and read as undefined value.</td></tr>
</table>
Correction:


<After Reset: Undefined>
<table>
<tr><th>Bit</th><th>Abbreviation</th><th>After Reset</th><th>R</th><th>W</th><th>Description</th></tr>
<tr><td>4</td><td>–</td><td>Undefined</td><td>?</td><td>0</td><td>Reserved Bit
Should be written with "0" and read as undefined value.</td></tr>
</table>
</td></tr>
<tr><td>Added in Rev. B</td><td>26-41</td><td>26.3.11 CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0 to 3)</td><td>
Setting value of the bit 7 to 0 (CiRFPCR bit) in the CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0 to 3) is corrected.
Error:
<table>
<tr><th>Bit</th><th>Abbreviation</th><th>After Reset</th><th>R</th><th>W</th><th>Description</th></tr>
<tr><td>7 to 0</td><td>CiRFPCR</td><td>Undefined</td><td>R</td><td>W</td><td>The CPU-side pointer for the receive FIFO is incremented by writing "H'FF"</td></tr>
</table>
Correction:
<table>
<tr><th>Bit</th><th>Abbreviation</th><th>After Reset</th><th>R</th><th>W</th><th>Description</th></tr>
<tr><td>7 to 0</td><td>CiRFPCR</td><td>Undefined</td><td>?</td><td>W</td><td>The CPU-side pointer for the receive FIFO is incremented by writing "H'FF"</td></tr>
</table>
</td></tr>
<tr><td>Added in Rev. B</td><td>26-45</td><td>26.3.13 CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0 to 3)</td><td>
Setting value of the bit 7 to 0 (CiTFPCR bit) in the CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0 to 3) is corrected.
Error:
<table>
<tr><th>Bit</th><th>Abbreviation</th><th>After Reset</th><th>R</th><th>W</th><th>Description</th></tr>
<tr><td>7 to 0</td><td>CiTFPCR</td><td>Undefined</td><td>R</td><td>W</td><td>The CPU-side pointer for the transmit FIFO is incremented by writing "H'FF"</td></tr>
</table>
Correction:
<table>
<tr><th>Bit</th><th>Abbreviation</th><th>After Reset</th><th>R</th><th>W</th><th>Description</th></tr>
<tr><td>7 to 0</td><td>CiTFPCR</td><td>Undefined</td><td>?</td><td>W</td><td>The CPU-side pointer for the transmit FIFO is incremented by writing "H'FF"</td></tr>
</table>
</td></tr>
</table>

Rev.	Page	Part	Contents
Added in Rev. B	26-56	26.3.20 CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0 to 4)	Description of the bit 7 (BLIF bit) in the CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0 to 4) is corrected. Error: **Bit** / **Abbreviation** / **After Reset** / **R** / **W** / **Description** 7 / BLIF / 0 / R / W / Bus Lock Detect Flag*1 The BLIF bit is set to "1" if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode. After the bit is set to "1", redetection takes place under either of the following conditions: • After this bit is set to "0" from "1", recessive bits are detected. • After this bit is set to "0" from "1", the CAN module enters CAN reset mode or CAN halt mode and then enters CAN operation mode again. 0: No bus lock detected 1: Bus lock detected Correction: **Bit** / **Abbreviation** / **After Reset** / **R** / **W** / **Description** 7 / BLIF / 0 / R / W / Bus Lock Detect Flag*1 The BLIF bit is set to "1" if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode. After the bit is set to "1", bus lock can be detected again after either of the following conditions is satisfied: • After this bit is set to "0" from "1", recessive bits are detected (bus lock is resolved). • After this bit is set to "0" from "1", the CAN module enters CAN reset mode and then enters CAN operation mode again (internal reset). 0: No bus lock detected 1: Bus lock detected

Rev.	Page	Part	Contents
Added in Rev. B	26-71	Figure 26.9 Transition between CAN Operating Modes (i = 0 to 3)	Figure 26.9 Transition between CAN Operating Modes (i = 0 to 3) : Incorrect description is corrected. Error:  CANM, SLPM, BOM, RBOC: Bits in the CiCTLR register Notes: *1 The transition timing from the bus-off state to CAN halt mode depends on the setting of the BOM bit. — When the BOM bit is "01", the state transition timing is immediately after entering the bus-off state. — When the BOM bit is "10", the state transition timing is at the end of the bus-off state. — When the BOM bit is "11", the state transition timing is at the setting of the CANM bit to "10" (CAN halt mode). *2 Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to "0" or "1". Correction:  CANM, SLPM, BOM, RBOC: Bits in the CiCTLR register Notes: *1 The transition timing from the bus-off state to CAN halt mode depends on the setting of the BOM bit. — When the BOM bit is "01", the state transition timing is immediately after entering the bus-off state. — When the BOM bit is "10", the state transition timing is at the end of the bus-off state. — When the BOM bit is "11", the state transition timing is at the setting of the CANM bit to "10" (CAN halt mode). *2 Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to "0" or "1". *3 The CAN module does not enter CAN halt mode while the CAN bus is locked in dominant state. Enter CAN reset mode instead.

Rev.	Page	Part	Contents
Added in Rev. B	26-73	Table 26.9 Operation in CAN Reset Mode and CAN Halt Mode	Table 26.9 Operation in CAN Reset Mode and CAN Halt Mode : Incorrect description is corrected.

Error:

Mode	Receiver	Transmitter	Bus-Off
CAN reset mode (forcible transition) CANM = "11"	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode without waiting for the end of message transmission.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN reset mode CANM = "01"	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode after waiting for the end of message transmission*1*4.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN halt mode	CAN module enters CAN halt mode after waiting for the end of message reception*2*3.	CAN module enters CAN halt mode after waiting for the end of message transmission*1*4.	[When the BOM bit is "00"] A halt request from a program will be acknowledged only after bus-off recovery. [When the BOM bit is "01"] CAN module enters automatically to CAN halt mode without waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bit is "10"] CAN module enters automatically to CAN halt mode after waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bit is "11"] CAN module enters CAN halt mode (without waiting for the end of bus-off recovery) if a halt is requested by a program during bus-off.

Legend: BOM bit: Bit in CiCTLR register (i = 0 to 3)

Notes: *1 If several messages are requested to be transmitted, mode transition occurs after the completion of the first transmission. In a case that the CAN reset mode is being requested during suspend transmission, mode transition occurs when the bus is idle, the next transmission ends, or the CAN module becomes a receiver.

*2 If the CAN bus is locked at the dominant level, the program can detect this state by monitoring the BLIF bit in the CiEIFR register.

*3 If a CAN bus error occurs during reception after CAN halt mode is requested, the CAN mode transits to CAN halt mode.

*4 If a CAN bus error or arbitration lost occurs during transmission after CAN reset mode or CAN halt mode is requested, the CAN mode transits to the requested CAN mode.

Correction:

Mode	Receiver	Transmitter	Bus-Off
CAN reset mode (forcible transition) CANM = "11"	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode without waiting for the end of message transmission.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN reset mode CANM = "01"	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode after waiting for the end of message transmission.*1*4.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN halt mode	CAN module enters CAN halt mode after waiting for the end of message reception.*2*3.	CAN module enters CAN halt mode after waiting for the end of message transmission.*1*2*4.	[When the BOM bit is "00"] A halt request from a program will be acknowledged only after bus-off recovery. [When the BOM bit is "01"] CAN module enters automatically to CAN halt mode without waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bit is "10"] CAN module enters automatically to CAN halt mode after waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bit is "11"] CAN module enters CAN halt mode (without waiting for the end of bus-off recovery) if a halt is requested by a program during bus-off.

Legend: BOM bit: Bit in CiCTLR register (i = 0 to 3)

Notes: *1 If several messages are requested to be transmitted, mode transition occurs after the completion of the first transmission. In a case that the CAN reset mode is being requested during suspend transmission, mode transition occurs when the bus is idle, the next transmission ends, or the CAN module becomes a receiver.

*2 If the CAN bus is locked in dominant state, the program can detect this state by monitoring the BLIF bit in the CiEIFR register. The CAN module does not enter CAN halt mode while the CAN bus is locked in dominant state. Enter CAN reset mode instead.

*3 If a CAN bus error occurs during reception after CAN halt mode is requested, the CAN module enters CAN halt mode. However, the CAN module does not enter CAN halt mode when the CAN bus is locked in dominant state.

*4 If a CAN bus error or arbitration lost occurs during transmission after CAN reset mode or CAN halt mode is requested, the CAN module enters the requested operating mode. However, the CAN module does not enter CAN halt mode when the CAN bus is locked in dominant state.

Rev.	Page	Part	Contents
Added in Rev. B	28-11	28.3 Register Descriptions	28.3 Register Descriptions : Incorrect description is corrected. Error: These flags are used to enable DMA transfer requests. Set these flags to "1" to enable a DMA transfer request, and set them to "0" to disable a request. To prevent incorrect DMA operation, only rewrite these bits from the DMA transfer masked state to the DMA transfer enabled state when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"). Do not rewrite from the DMA transfer enabled state to the DMA transfer masked state when DRI acquisition is enabled, since that can result in a DMA request not being handled. Correction: These flags are used to enable DMA transfer requests. Set these flags to "1" to enable a DMA transfer request, and set them to "0" to disable a DMA transfer request. To prevent incorrect DMA operation, only rewrite these bits from the DMA transfer request masked state to the DMA transfer request enabled state when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"). Do not rewrite from the DMA transfer request enabled state to the DMA transfer request masked state when DRI acquisition is enabled, since that can result in a DMA transfer request not being handled.
Added in Rev. B	28-16	28.3.4 DRI0DIN DMA Transfer Enable Register (DRI0DINDEN)	28.3.4 DRI0DIN DMA Transfer Enable Register (DRI0DINDEN) : Incorrect description is corrected. Error: Also note that it is only possible to rewrite the DRI0DINDEN register bits from the transfer masked state to the transfer enabled state when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"). Do not rewrite from the transfer enabled state to the transfer masked state when DRI acquisition is enabled. Correction: Also note that it is only possible to rewrite the DRI0DINDEN register bits from the DMA transfer request masked state to the DMA transfer request enabled state when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"). Do not rewrite from the DMA transfer request enabled state to the DMA transfer request masked state when DRI acquisition is enabled.
Added in Rev. B	28-21	28.3.8 DRI0DEC DMA Transfer Enable Register (DRI0DECDEN)	28.3.8 DRI0DEC DMA Transfer Enable Register (DRI0DECDEN) : Incorrect description is corrected. Error: If a DMA transfer request mask (disable) setting and an internal DMA transfer request occur at the same time, the DMA transfer request mask (disable) setting takes precedence. Also note that it is only possible to rewrite the DRI0DECDEN register bits from the transfer masked state to the transfer enabled state when DEC counter operation is enabled (DRIiDECnCNT.DECnEN bit = "1"). Do not rewrite from the transfer enabled state to the transfer masked state when DEC counter operation is enabled. Correction: If a DMA transfer request mask (disable) setting and an internal DMA transfer request occur at the same time, the DMA transfer request mask (disable) setting takes precedence. Also note that it is only possible to rewrite the DRI0DECDEN register bits from the DMA transfer request masked state to the DMA transfer request enabled state when DEC counter operation is enabled (DRIiDECnCNT.DECnEN bit = "1"). Do not rewrite from the DMA transfer request enabled state to the DMA transfer request masked state when DEC counter operation is enabled.

<table>
<tr><th>Rev.</th><th>Page</th><th>Part</th><th>Contents</th></tr>
<tr><td>Added in Rev. B</td><td>28-27</td><td>28.3.12 DRI0 DMA Transfer Enable Register (DRI0TRMDEN)</td><td>28.3.12 DRI0 DMA Transfer Enable Register (DRI0TRMDEN) : Incorrect description is corrected.
Error: Controls the enabled/disabled states for DRI0 transfer related DMA transfer requests. If one of these bits is set to "1", the corresponding DMA transfer request signal output is enabled. If a DMA transfer mask (disable) is set at the same time as an internal DMA transfer request, the DMA transfer mask (disable) takes precedence. Also note that when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"), the DRI0TRMDEN register may only be rewritten from the transfer masked state to the transfer enabled state. Do not rewrite any bits in this register from the transfer enabled state to the transfer masked state when DRI acquisition is enabled.
Correction: Controls the enabled/disabled states for DRI0 transfer related DMA transfer requests. If one of these bits is set to "1", the corresponding DMA transfer request signal output is enabled. If a DMA transfer mask (disable) is set at the same time as an internal DMA transfer request, the DMA transfer mask (disable) takes precedence. Also note that when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"), the DRI0TRMDEN register may only be rewritten from the DMA transfer request masked state to the DMA transfer request enabled state. Do not rewrite any bits in this register from the DMA transfer request enabled state to the DMA transfer request masked state when DRI acquisition is enabled.</td></tr>
<tr><td>Added in Rev. B</td><td>32-13</td><td>32.4.1 FlexRay Operation Control Register (FXROC)</td><td>Description of the bit 2 (FBSEN bit) in the FlexRay Operation Control Register (FXROC) is corrected.
Error: - FRNVMn
- FRNVMn
Correction: - FRNMVn
- FRNMVn</td></tr>
<tr><td>Added in Rev. B</td><td>32-139</td><td>32.12.5 Configuration of NIT Start and Offset Correction Start</td><td>32.12.5 Configuration of NIT Start and Offset Correction Start : Incorrect description is corrected.
Error: For the FlexRay module the offset correction start is required to be the OCS bit in the FRGTUC4 register σ the NIT bit int the FRGTUC4 register + 1 = k+1.
Correction: For the FlexRay module the offset correction start is required to be the OCS bit in the FRGTUC4 register ≥ the NIT bit in the FRGTUC4 register + 1 = k+1.</td></tr>
<tr><td>Added in Rev. B</td><td>32-145</td><td>Table 32.8 State Transitions of FlexRay overall state Machine</td><td>Table 32.8 State Transitions of FlexRay overall state Machine : Incorrect description is corrected.
Error:
<table>
<tr><th>T#</th><th>Condition</th><th>From</th><th>To</th></tr>
<tr><td>T1</td><td>Hard reset</td><td>All states</td><td>DEFALT_CONFIG</td></tr>
<tr><td>T2</td><td>Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001</td><td>DEFALT_CONFIG</td><td>CONFIG</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>T15</td><td>Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001</td><td>HALT</td><td>DEFALT_CONFIG</td></tr>
</table>
Correction:
<table>
<tr><th>T#</th><th>Condition</th><th>From</th><th>To</th></tr>
<tr><td>T1</td><td>Hard reset</td><td>All states</td><td>DEFAULT_CONFIG</td></tr>
<tr><td>T2</td><td>Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001</td><td>DEFAULT_CONFIG</td><td>CONFIG</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>T15</td><td>Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001</td><td>HALT</td><td>DEFAULT_CONFIG</td></tr>
</table></td></tr>
</table>