RENESAS TECHNICAL UPDATE

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

Product Category	MPU & MCU		Document No.	TN-16C-A235A/E	Rev.	1.00
Title	Errata to R32C/151 Group User's Manual: Hardware		Information Category	Technical Notification		
Applicable Product	R32C/151 Group	Lot No.	Reference Document	R32C/151 Group User's Manual: Hardware Rev. 1.10 (REJ09B0503-0110)		

This document describes corrections to the R32C/151 Group User's Manual: Hardware, Rev. 1.10.
The corrections are indicated in red in the list below.

- Page 16 of 628, register symbol "R3R0" in line 3 of 2.1.1 is corrected as follows:
  "R3R1"

- Page 34 of 628, register name "Increment/Decrement Counting Select Register" in Table 4.15 is corrected as follows:
  "Increment/Decrement Select Register"

- Pages 61, 62, 75, and 76 of 628, register name "CAN1/0 Acceptance Mask Register 0/1/2/3/4/5/6/7" in **Tables 4.42, 4.43, 4.56, and 4.57** is corrected as follows:
  "CAN1/0 Mask Register 0/1/2/3/4/5/6/7"

- Pages 64 and 78 of 628, reset value "XXXX XX00b" for registers C1MSMR and C0MSMR in **Tables 4.45 and4.59** is corrected as follows:
  "0000 0000b"

- Page 88 of 628, description of the third paragraph of 6.2.1 is modified as follows:
  "When the voltage rises to or above Vdet(R) again, the VMF bit becomes 1 (VCC ≥ Vdet) and the LVDF bit becomes 1. At this point, an interrupt request is generated when the LVDIEN bit is 1."

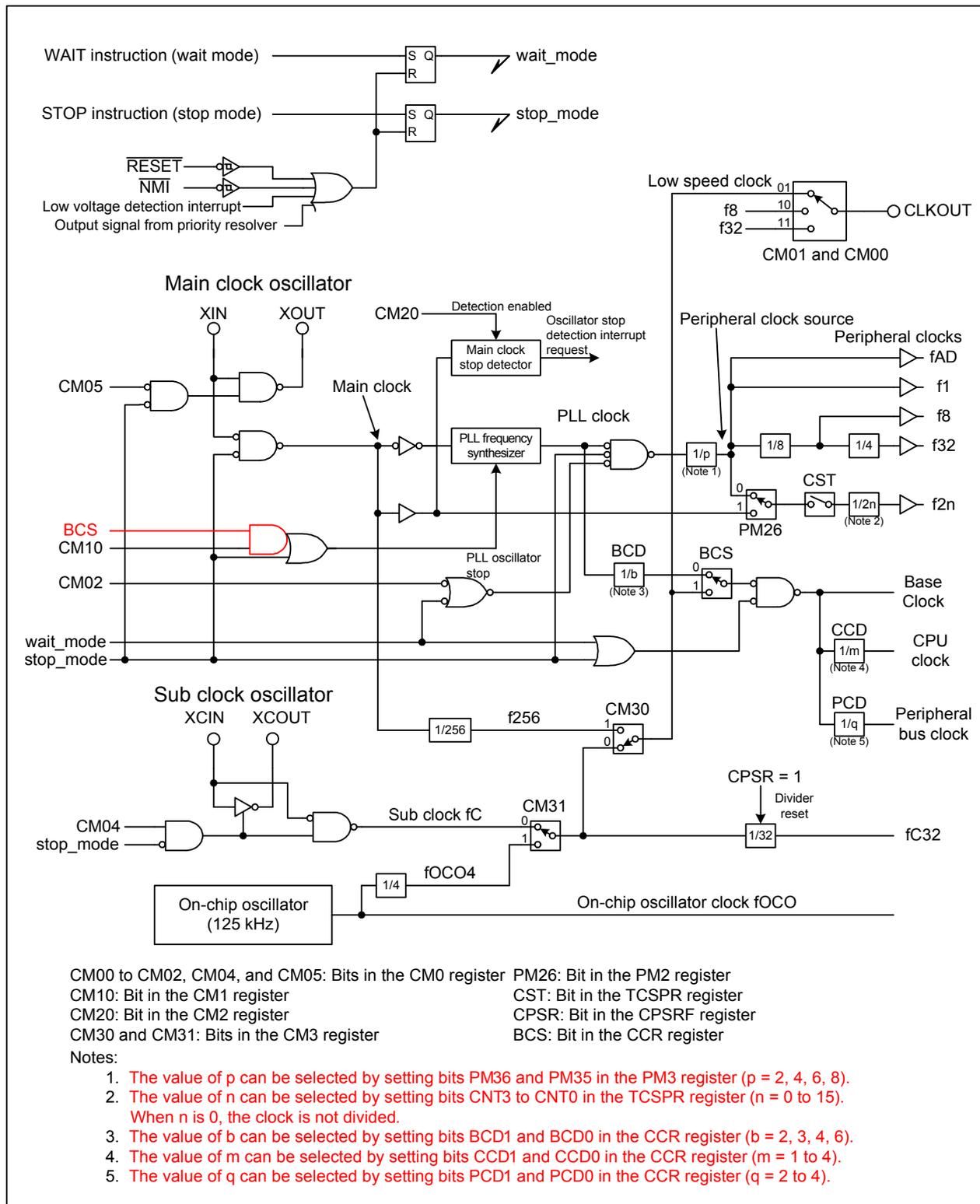•Page 91 of 628, Figure 7.1 is corrected as follows:

Figure 7.1   Clock Generation Circuitry

•Page 93 of 628, bit name "XCIN-XCOUT Drive Power Select Bit" in Figure 7.3 is modified as follows:
  "XCIN-XCOUT Drive Strength Select Bit"

• Page 94 of 628, bit name "XIN-XOUT Drive Power Select Bit" in Figure 7.4 is modified as follows:
"XIN-XOUT Drive Strength Select Bit"

• Page 94 of 628, description of Note 2 in Figure 7.4 is modified as follows:
"When the BCS bit in the CCR register is 0 (PLL clock selected as base clock source), the PLL frequency synthesizer does not stop oscillating even if the CM10 bit is set to 1."

• Page 94 of 628, bit symbol "CM02" in Note 3 of Figure 7.5 is corrected as follows:
"CM20"

• Page 104 of 628, description of the last paragraph of 7.1.4 is modified as follows:
"It starts running when setting any of the following bits to 1; the CM31 bit in the CM3 register or the PM22 bit in the PM2 register. It is not necessary to wait for stabilization because the on-chip oscillator instantly starts oscillating."

• Page 105 of 628, description "(Refer to Figure 7.18 "State Transition (when the sub clock is used)")" is deleted from 7.2.

• Page 114 of 628, description of 7.7.2 is modified as follows:
"The base clock stops in wait mode, so clocks generated by the base clock, the CPU clock and peripheral bus clock, stop running as well. Thus the CPU and watchdog timer, operated by these two clocks, also stop."

• Page 123 of 628, "IOBC" is added to description in Function of the PRC1 bit in Figure 9.1 as follows:
"Enable writing to registers PM0, PM2, INVC0, INVC1, and IOBC"

• Page 123 of 628, description "Bits PRC0 and PRC1 do not automatically become 0. They should be set to 0 by a program." is deleted from Note 1 of Figure 9.1.

• Page 127 of 628, description of Note 1 in Figure 10.1 is modified as follows:
"The peripheral interrupts are generated by the corresponding peripherals in the MCU."

• Page 128 of 628, the second paragraph of (5) in 10.2 is modified as follows:
"The stack pointer (SP) used for this interrupt differs depending on the software interrupt numbers. For software interrupt numbers 0 to 127, when an interrupt request is accepted, the U flag is saved and set to 0 to select the interrupt stack pointer (ISP) during the interrupt sequence. The saved data of the U flag is restored upon returning from the interrupt handler. For software interrupt numbers 128 to 255, the stack pointer does not change during the interrupt sequence."

• Page 130 of 628, description of 10.5 is corrected as follows:
"Each interrupt vector has a 4-byte memory space, in which the start address of the associated interrupt handler is stored. When an interrupt request is accepted, a jump to the address set in the interrupt vector takes place. Figure 10.2 shows an interrupt vector."

• Page 131 of 628, description in the Remarks for the BRK instruction in Table 10.1 is corrected as follows:
"If address FFFFFFE7h is FFh, a jump to the interrupt vector of software interrupt number 0 in the relocatable vector table takes place"

• Page 138 of 628, description for the IR bit below Figure 10.4 is corrected as follows:
"The IR bit becomes 1 (interrupt requested) when an interrupt request is generated; this bit setting is retained until the interrupt request is accepted. When the request is accepted and a jump to the corresponding interrupt vector takes place, the IR bit becomes 0 (no interrupt requested).
The IR bit can be set to 0 by a program. This bit should not be set to 1."

•Page 142 of 628, description of Note 1 in Table 10.7 is corrected as follows:
"These are the values when the interrupt vectors are aligned to the addresses in multiples of 4 in the internal ROM. However, the condition does not apply to the fast interrupt."

•Page 144 of 628, description in lines 4 to 5 and 7 to 8 of 10.8 as follows:
"For maskable interrupts (peripheral interrupts), the interrupt request level select bits (bits ILVL2 to ILVL0) select a request level. If two or more interrupt requests have the same request level, the interrupt with higher priority, predetermined by hardware, is accepted.
The priorities of the reset and special interrupts, such as the watchdog timer interrupt, are determined by the hardware. Note that the reset has the highest priority. The following is the priority order determined by the hardware:"

•Page 151 of 628, register symbol "IIOiE" in line 16 of 10.13 is corrected as follows:
"IIOiIE"

•Page 155 of 628, description in lines 4 to 6 of 11. Watchdog Timer is corrected as follows:
"Select either an interrupt request or a reset with the CM06 bit in the CM0 register for when the watchdog timer underflows. Once the CM06 bit is set to 1 (reset), it cannot be changed to 0 (watchdog timer interrupt) by a program. It can be set to 0 only by a reset."

•Page 155 of 628, register symbol "WKD" of 11. Watchdog Timer is corrected as follows:
"WDK"

•Page 160 of 628, expression "a value more than 00000001h" in the Specification of DMA transfer start-up in Table 12.1 is corrected as follows:
"a value other than 00000000h"

•Page 168 of 628, description of the first paragraph in 12.1 is corrected as follows:
"The transfer cycle is composed of bus cycles to read data from (source read) or to write data to (destination write) memory or an SFR."

•Page 174 of 628, source address "FFFFFFFh" in Note 1 of Table 13.1 is corrected as follows:
"FFFFFFFFh"

•Page 174 of 628, bit symbol "IIRLT" in the fifth bullet point of 13.1 is corrected as follows:
"IRLT"

•Pages 176 and 177 of 628, expression "DMA II transfer complete interrupt vector address" in lines 3 to 4 and the seventh bullet point of 13.1.2 and Figure 13.2 is corrected as follows:
"jump address for the DMA II transfer complete interrupt handler"

•Pages 176 and 179 of 628, expression "interrupt vector" in Figure 13.2 and 13.1.4 is corrected as follows:
"interrupt vector space"

•Page 177 of 628, expression "jump address" in the seventh bullet point of 13.1.2 is corrected as follows:
"start address"

•Page 178 of 628, bit names of the OPER bit and bits CNT0 to CNT2 in Figure 13.3 are modified as follows:
OPER: "Calculation Result Transfer Select Bit"
CNT0 to CNT2: "Number of Transfers Setting Bit"

•Page 188 of 628, description of the third bullet point of 15.1 is corrected as follows:
"One-shot timer mode: The timer outputs pulses after a trigger input until the counter reaches 0000h"

- Page 191 of 628, "Counting" is deleted from bit names of bits TA0UD to TA4UD "Timer A0/1/2/3/4 Increment/ Decrement Counting Select Bit" and the register name "Increment/Decrement Counting Select Register" in Figure 15.7.

- Page 193 of 628, descriptions "b2 b3", "b4 b5", and "b6 b7" in Figure 15.9 are corrected as follows: "b3 b2", "b5 b4", and "b7 b6"

- Page 199 of 628, bit name of the MR2 bit in Figure 15.12 is modified as follows: "Increment/Decrement Switching Source Select Bit"

- Page 199 of 628, bit symbols "TAiTGH and TAiTGL" in Note 5 of Figure 15.12 are corrected as follows: "TAjTGH and TAjTGL"

- Page 200 of 628, pin name "INT2" in Figures 15.13 and 15.14 is corrected as follows: "INT2"

- Page 204 of 628, bit symbol "TAiS" in Function of the MR2 bit in Figures 15.16 is corrected as follows: "TAiOS"

- Page 219 of 628, description of 15.3.3.1 is corrected as follows: "While the timer counter is running, the TBj register indicates a counter value at any given time (j = 0 to 5). However, FFFFh is read while reloading is in progress. When a value is set to the TBj register while the timer counter is stopped, if the TBj register is read before the count starts, the set value is read."

- Page 219 of 628, expression "TBj interrupt handler" in the eighth bullet point of 15.3.3.2 is changed as follows: "timer Bj interrupt handler"

- Page 223 of 628, description of Note 1 in Figure 16.3 is modified as follows: "Set this register after setting the PRC1 bit in the PRCR register to 1 (write enabled). Also, rewrite this register while timers A1, A2, A4, and B2 are stopped."

- Page 230 of 628, description "The sum of setting values for registers TAi and TAi1 should be identical to the setting value of the TB2 register in this mode." is deleted from lines 8 to 9 of 16.3.

- Page 230 of 628, description in line 11 of 16.3 is corrected as follows: "Figure 16.11 shows registers TA1M, TA2M, TA4M, TA11M, TA21M, and TA41M in this function."

- Page 235 of 628, bit symbol "INV06" in Note 3 of Figure 16.16 is corrected as follows: "INV16"

- Page 236 of 628, register symbol "INV1" in Note 2 of Figure 16.18 is corrected as follows: "INVC1"

- Pages 244 and 245 of 628, descriptions for the CRD bit in Figures 17.5 and 17.6 are modified as follows:

Bit Symbol	Bit Name	Function	RW
CRD	CTS Function Disable Bit	0: CTS function enabled 1: CTS function disabled	RW

- Page 246 of 628, description of function of the UiIRS bit in Figure 17.7 is modified as follows:
  "0: Transmit buffer is empty (TI = 1)
  1: Transmission is completed (TXEPT = 1)"

•Page 246 of 628, bit name "Logical Inversion Select Bit" in Figure 17.7 is modified as follows:
  "Logic Inversion Select Bit"

•Page 248 of 628, bit name "Clock Synchronous Bit" in Figure 17.11 is modified as follows:
  "Clock Synchronization Bit"

•Page 249 of 628, expression "UiBRG count source" in the function of bits DL0 to DL2 in Figure 17.12 is corrected as follows:
  "baud rate generator count source"

•Page 250 of 628, bit name of the RSTAREQ bit in Figure 17.13 is modified as follows:
  "Repeated START Condition Generate Bit"

•Page 256 of 628, "TXEPT flag" in Figure 17.18 is corrected as follows:
  "TXEPT bit"

•Page 256 of 628, description in the fourth dash of Figure 17.18 is corrected as follows:
  "The UiIRS bit in registers UiC1 and U34CON is 0 (an interrupt request is generated when the transmit buffer is empty)"

•Pages 264 and 265 of 628, descriptions of function of the UiIRS bit in Figures 17.23 and 17.24 are corrected as follows:
  1: "(an interrupt request is generated when transmission is completed)"
  0: "(an interrupt request is generated when the transmit buffer is empty)"

•Page 301 of 628, description of 18.1.5 is modified as follows:
  "In repeat sweep mode 1, the analog voltage applied to eight selected pins including one to four prioritized pins is repeatedly converted into a digital code. Table 18.6 lists specifications of repeat sweep mode 1."

•Page 301 of 628, description in the specification of the function in Table 18.6 is modified as follows:
  "The analog voltage applied to eight selected pins including one to four prioritized pins is repeatedly converted into a digital code. The prioritized pins are selected by setting bits SCAN1 and SCAN0 in the AD0CON1 register and bits APS1 and APS0 in the AD0CON2 register"

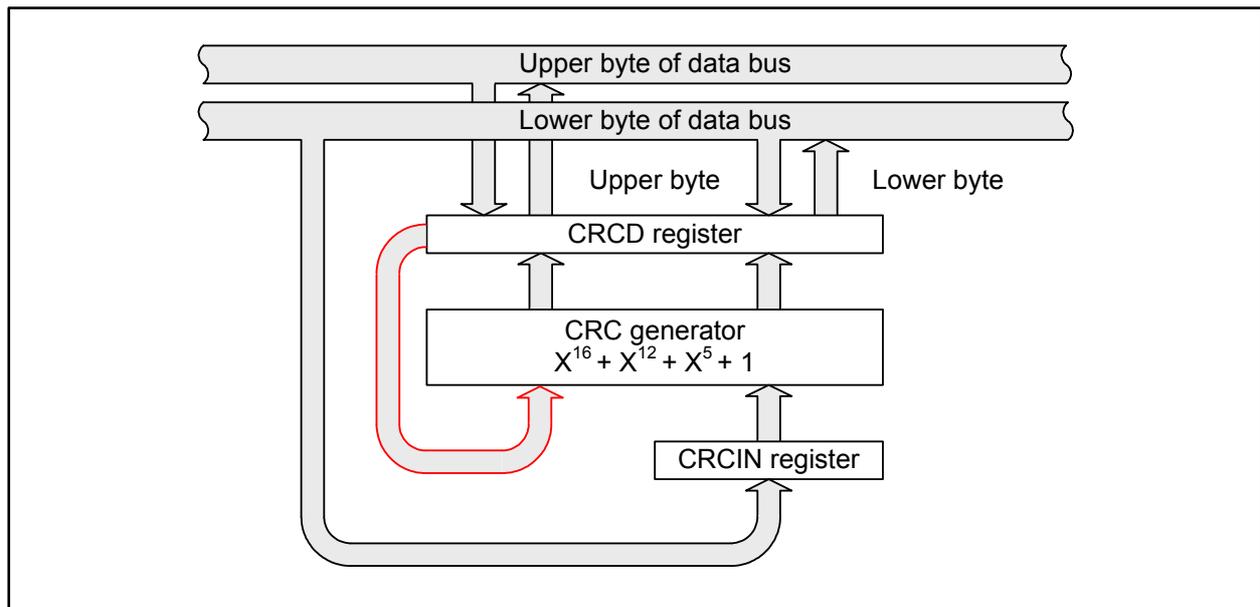•Page 313 of 628, Figure 20.1 is corrected as follows:

Figure 20.1   CRC Calculator Block Diagram

•Page 327 of 628, descriptions for bits UD0 and UD1 in Figure 22.7 are modified as follows:

Bit Symbol	Bit Name	Function	RW
UD0	Increment/Decrement Control Bit	b6 b5 0 0 : Increment mode 0 1 : Increment/decrement mode	RW
UD1		1 0 : Two-phase pulse signal processing mode [6] 1 1 : Do not use this combination	RW

•Page 328 of 628, Note 3 "The GOC bit becomes 0 after gating is cleared." is deleted from Figure 22.8.

•Page 335 of 628, description in the second bullet point of the reset conditions in Table 22.2 is modified as follows:
    "An input of low signal into the external interrupt pin ($\overline{\text{INT0}}$ or $\overline{\text{INT1}}$) as follows:"

•Page 335 of 628, description of the first bullet point of the selectable functions in Table 22.2 is corrected as follows:
    "The base timer starts counting when the BTS or BTiS bit is set to 1. When the base timer reaches FFFFh, it starts decrementing. When the RST1 bit in the GiBCR1 register is 1 (the base timer is reset by matching with the GiPO0 register), the timer counter starts decrementing two counts after the base timer value matches the GiPO0 register setting. When the timer counter reaches 0000h, it starts incrementing again (refer to Figure 22.20)."

•Pages 346, 348, and 351 of 628, expression "Input to the IIOi_j pin" in Figures 22.24 to 22.26 is corrected as follows:
    "IIOi_j pin"

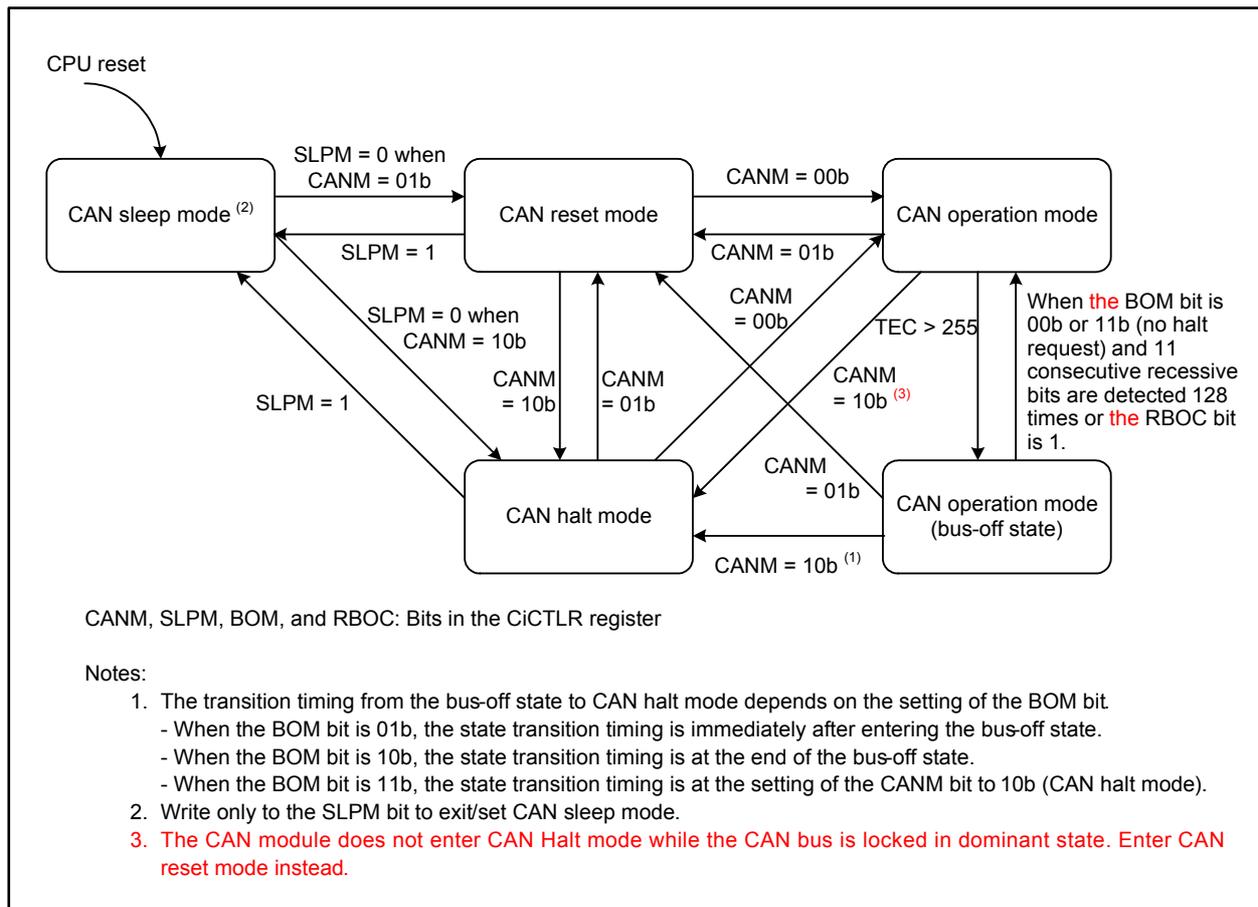•Page 362 of 628, expression "b6 b5 b4" in Figures 23.3 is corrected as follows:
    "b2 b1 b0"

- Page 369 of 628, description "write to this register in 16-bit units" in Figures 23.11 is corrected as follows:
  "read this register in 16-bit units"

- Page 370 of 628, description "with bits CKS2 to CKS0" to the first paragraph of 23.1.1 is added as follows:
  "When the MSS bit in the SSiCRH register is 1 (master mode), select a transmit/receive clock from among seven internal clocks (f(BCLK) divided-by-4, -8, -16, -32, -64, -128, and -256) with bits CKS2 to CKS0."

- Page 384 of 628, bit symbol "MSL" in line 14 of 23.1.7 is corrected as follows:
  "MLS"

- Page 432 of 628, descriptions for the RBOC bit in Figure 25.2 are modified as follows:

Bit Symbol	Bit Name	Function	RW
RBOC	Forced Recovery From Bus-off Bit [4]	0: Nothing occurred 1: Forced recovery from bus-off [5]	RW

- Page 439 of 628, description in 25.1.4.1 is corrected as follows:
  "When the EID bit is 0, the corresponding EID bit in a received message is not compared.
  When this bit is 1, the corresponding EID bit in a received message is compared."

- Page 439 of 628, description in 25.1.4.2 is corrected as follows:
  "When the SID bit is 0, the corresponding SID bit in a received message is not compared.
  When this bit is 1, the corresponding SID bit in a received message is compared."

- Page 449 of 628, expression "fCAN (CAN system clock)" in line 5 of 25.1.9.5 is corrected as follows:
  "the peripheral bus clock"

- Page 452 of 628, expression "fCAN" in line 6 of 25.1.10.3 is corrected as follows:
  "the peripheral bus clock"

- Page 458 of 628, description of function of b7 in Figure 25.17 is corrected as follows:
  b7: "No register bit; the read value is 0"

- Page 462 of 628, description of function of b6-b5 in Figure 25.19 is corrected as follows:
  b6-b5: "No register bits; the read value is 0"

- Page 470 of 628, description in 25.1.20.8 is corrected as follows (refer to TN-16C-A229A/E):
  "The BLIF bit becomes 1 if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode.
  After the BLIF bit becomes 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)."

•Page 479 of 628, Note 3 is added to Figure 25.34 as follows (refer to TN-16C-A229A/E):

CANM, SLPM, BOM, and 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 01b, the state transition timing is immediately after entering the bus-off state.
   - When the BOM bit is 10b, the state transition timing is at the end of the bus-off state.
   - When the BOM bit is 11b, the state transition timing is at the setting of the CANM bit to 10b (CAN halt mode).
2. Write only to the SLPM bit to exit/set CAN sleep mode.
3. The CAN module does not enter CAN Halt mode while the CAN bus is locked in dominant state. Enter CAN reset mode instead.

• Page 481 of 628, Table 25.9 is corrected as follows (refer to TN-16C-A229A/E):

Table 25.9   Operation in CAN Reset Mode and CAN Halt Mode

Mode	Receiver	Transmitter	Bus-off
CAN reset mode	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 00b A halt request from a program will be acknowledged only after bus-off recovery - When the BOM bit is 01b CAN module automatically enters 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 10b CAN module automatically enters 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 11b 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

BOM bit: Bit in the CiCTLR register (i = 0, 1)

Notes:
1. If several messages are requested to be transmitted, mode transition occurs after the completion of the first message transmission. When 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.

• Page 482 of 628, description of the first paragraph in 25.2.3 is modified as follows:
   "CAN sleep mode is used for reducing current consumption by stopping the clock supply to the CAN module. After a MCU reset, the CAN module starts from CAN sleep mode."

• Page 485 of 628, q value "q = 1, 2, 3, 4" in Figure 25.36 is corrected as follows:
   "q = 2, 3, 4"

• Page 497 of 628, description of the first and second paragraphs in 26. I/O Pins is corrected as follows (refer to TN-16C-A200A/E):
   "Each pin of the MCU functions as a programmable I/O port or an I/O pin for integrated peripherals. These functions can be switched by the function select registers. The pull-up resistors are enabled for every group of four pins. However, when a pin functions as an output pin, a pull-up resistor is disabled regardless of the register settings."

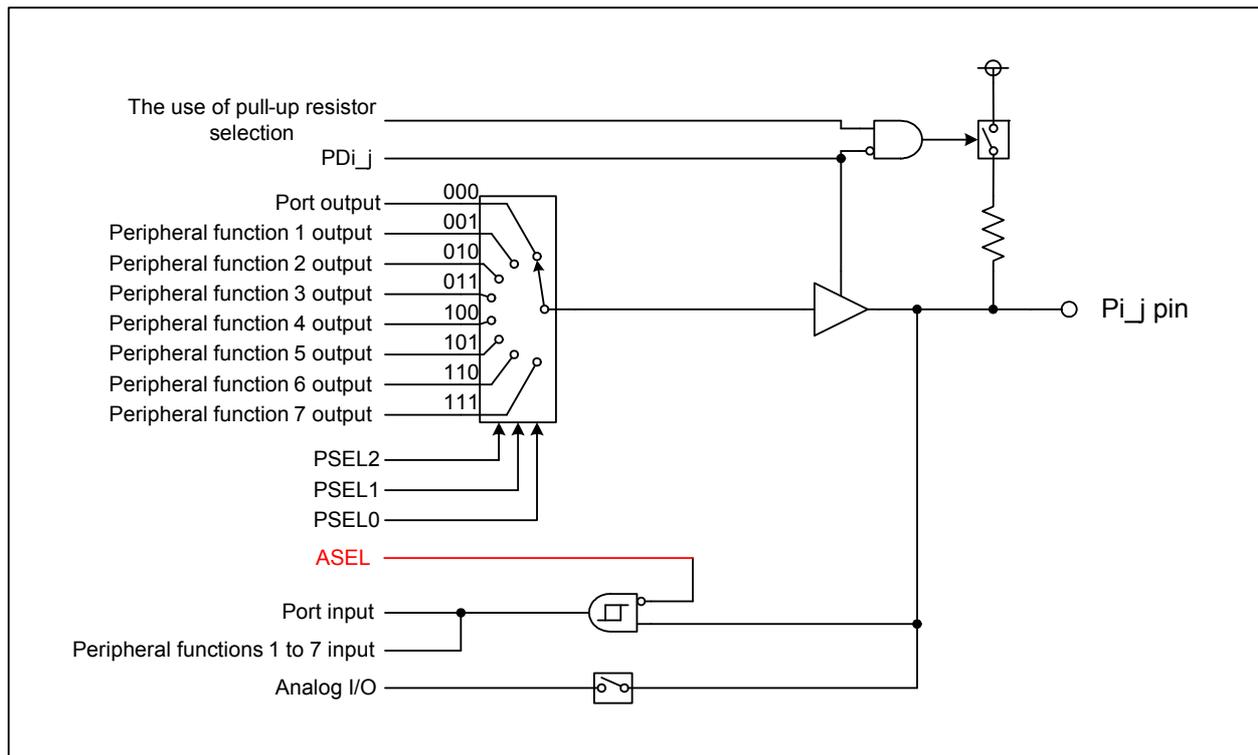•Page 497 of 628, Figure 256.1 is corrected as follows (refer to TN-16C-A200A/E):

Figure 26.1　Typical I/O Pin Block Diagram (i = 0 to 15; j = 0 to 7)

•Page 497 of 628, description in the last paragraph of 26. I/O Pins is corrected as follows:
"The input-only port P8_5 shares a pin with NMI and has neither function select register nor the corresponding direction bit. Port P14_1 also functions as an input-only port. The function select register and bit 1 in the PD14 register are reserved. Port P9 is protected from unexpected write accesses by the PRC2 bit in the PRCR register. Ports P3, P7, and P8 are protected from unexpected write accesses by the PRC30 bit in the PRCR3 register (refer to 9. "Protection")."

•Pages 501, 504, and 505 of 628, expressions "IIO0 output" and "IIO1 output" in Figures 26.4, 26.7, and 26.8 are changed as follows:
"IIO0_i output" and "IIO1_i output"

•Page 503 of 628, "PD3_i register" in line 4 of the paragraph below Figure 26.6 is corrected as follows:
"PD3_i bit"

•Page 545 of 628, Note 2 of Table 27.12 is corrected as follows:
"The program is performed in 64-bit (4-word) units. A sequence of commands consists of commands from the second to fifth. The upper 29 bits of the address WA should be fixed and the lower 3 bits of respective commands from the second to fifth should be set to 000b, 010b, 100b, and 110b for the addresses 0h, 2h, 4h, and 6h, or 8h, Ah, Ch, and Eh."

•Page 551 of 628, description in 27.3.4.7 is corrected as follows:
"Execute this command to verify if a specified block in the flash memory is locked. This command can be used when the LBM bit in the FMR0 register is 1 (read by the LBS bit).
The LBS bit in the FMSR0 register reflects the lock bit status of the specified block when the following is performed: first write 0071h to address FFFFF800h and verify that the FCA bit in the FMR0 register becomes 0. Then write 00D0h to an even address of the corresponding block.
Read the LBS bit after the RDY bit in the FMSR0 register becomes 1 (ready)."

• Page 559 of 628, description in 27.6.7 is corrected as follows:
"The time to execute software commands (program, block erase, lock bit program, and protect bit program) increases as the number of program/erase cycles increases. If the number of program/erase cycles exceeds the endurance value specified in the electrical characteristics, it may take an unpredictable amount of time to execute the software commands. The wait time for executing software commands should be set much longer than the execution time specified in the electrical characteristics."

• Page 559 of 628, description in the first bullet point of 27.6.8 is corrected as follows:
"The minimum values of program/erase cycles specified in the electrical characteristics are the maximum values that can guarantee the initial performance of the flash memory. The program/erase operation may still be performed even if the number of program/erase cycles exceeds the guaranteed values."

• Pages 576 and 577 of 628, "Programming and erasure endurance of flash memory" in Tables 29.8 and 29.9 is changed as follows:
"Program/erase cycles"

• Pages 576 and 577 of 628, unit "times" in Tables 29.8 and 29.9 is corrected as follows:
"Cycles"

• Pages 583 and 594 of 628, expression "Driver power" in Tables 29.18 and 29.38 is modified as follows:
"Drive strength"

• Page 607 of 628, "counting" is deleted from the UDF register name "Increment/decrement counting select register" in Table 30.1.

• Page 614 of 628, description of 30.6.3.1 is corrected as follows:
"While the timer counter is running, the TBj register indicates a counter value at any given time (j = 0 to 5). However, FFFFh is read while reloading is in progress. When a value is set to the TBj register while the timer counter is stopped, if the TBj register is read before the count starts, the set value is read."

• Page 614 of 628, expression "TBj interrupt handler" in the eighth bullet point of 30.6.3.2 is changed as follows:
"timer Bj interrupt handler"

• Page 616 of 628, expression "restart condition" in 30.8.3 is changed as follows:
"repeated START condition"

• Page 622 of 628, description in 30.11.7 is corrected as follows:
"The time to execute software commands (program, block erase, lock bit program, and protect bit program) increases as the number of program/erase cycles increases. If the number of program/erase cycles exceeds the endurance value specified in the electrical characteristics, it may take an unpredictable amount of time to execute the software commands. The wait time for executing software commands should be set much longer than the execution time specified in the electrical characteristics."

• Page 622 of 628, description in the first bullet point of 30.11.8 is corrected as follows:
"The minimum values of program/erase cycles specified in the electrical characteristics are the maximum values that can guarantee the initial performance of the flash memory. The program/erase operation may still be performed even if the number of program/erase cycles exceeds the guaranteed values."