Date: Nov. 1, 2010

RENESAS TECHNICAL UPDATE

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

Product Category	MPU & MCU		Document No.	TN-16C-A193A/E	Rev.	1.00
Title	Errata to R32C/151 Group Hardware Manual		Information Category	Technical Notification		
Applicable Product	R32C/151 Group	Lot No.	Reference Document	R32C/151 Group Hardware Manual Rev. 1.03 (REJ09B0503-0103)		

This document describes corrections to the R32C/151 Group Hardware Manual, Rev. 1.03.
The corrections are indicated in red in the list below.

- Page 12 of 624, description “Output of the clock with the same frequency as fC, f8, or f32” for Clock output in Table 1.8 is corrected as follows:
  “Output of the clock with the same frequency as low speed clocks, f8, or f32”

- Page 25 of 624, description of register name “Group 1 Timer Measurement Prescaler Register 6/7” in Table 4.6 is corrected as follows:
  “Group 1 Time Measurement Prescaler Register 6/7”

- Page 28 of 624, description of register name “Group 0 Timer Measurement Prescaler Register 6/7” in Table 4.9 is corrected as follows:
  “Group 0 Time Measurement Prescaler Register 6/7”

- Page 31 of 624, description of register name “Group 3 Timer Measurement Prescaler Register 6/7” in Table 4.12 is corrected as follows:
  “Group 3 Time Measurement Prescaler Register 6/7”

- Page 34 of 624, description of register name “UART2 Transmission/Receive Mode Register” in Table 4.15 is corrected as follows:
  “UART2 Transmit/Receive Mode Register”

- Page 45 of 624, description of register name “External Interrupt Source Select Register 1/0” in Table 4.26 is corrected as follows:
  “External Interrupt Request Source Select Register 1/0”

- Page 64 of 624, descriptions of register names “CAN1 Reception Error Count Register” and “CAN1 Transmission Error Count Register” in Table 4.45 are corrected as follows:
  “CAN1 Receive Error Count Register” and “CAN1 Transmit Error Count Register”

- Page 78 of 624, descriptions of register names “CAN0 Reception Error Count Register” and “CAN0 Transmission Error Count Register” in Table 4.59 are corrected as follows:
  “CAN0 Receive Error Count Register” and “CAN0 Transmit Error Count Register”

•Page 86 of 624, descriptions for the VDEN bit in Figure 6.4 are corrected as follows:

Bit Symbol	Bit Name	Function	RW
VDEN	Low Voltage Detector Enable Bit	0: Low voltage detector disabled 1: Low voltage detector enabled	RW

•Page 86 of 624, descriptions of Notes 2 and 3 in Figure 6.4 are corrected as follows:

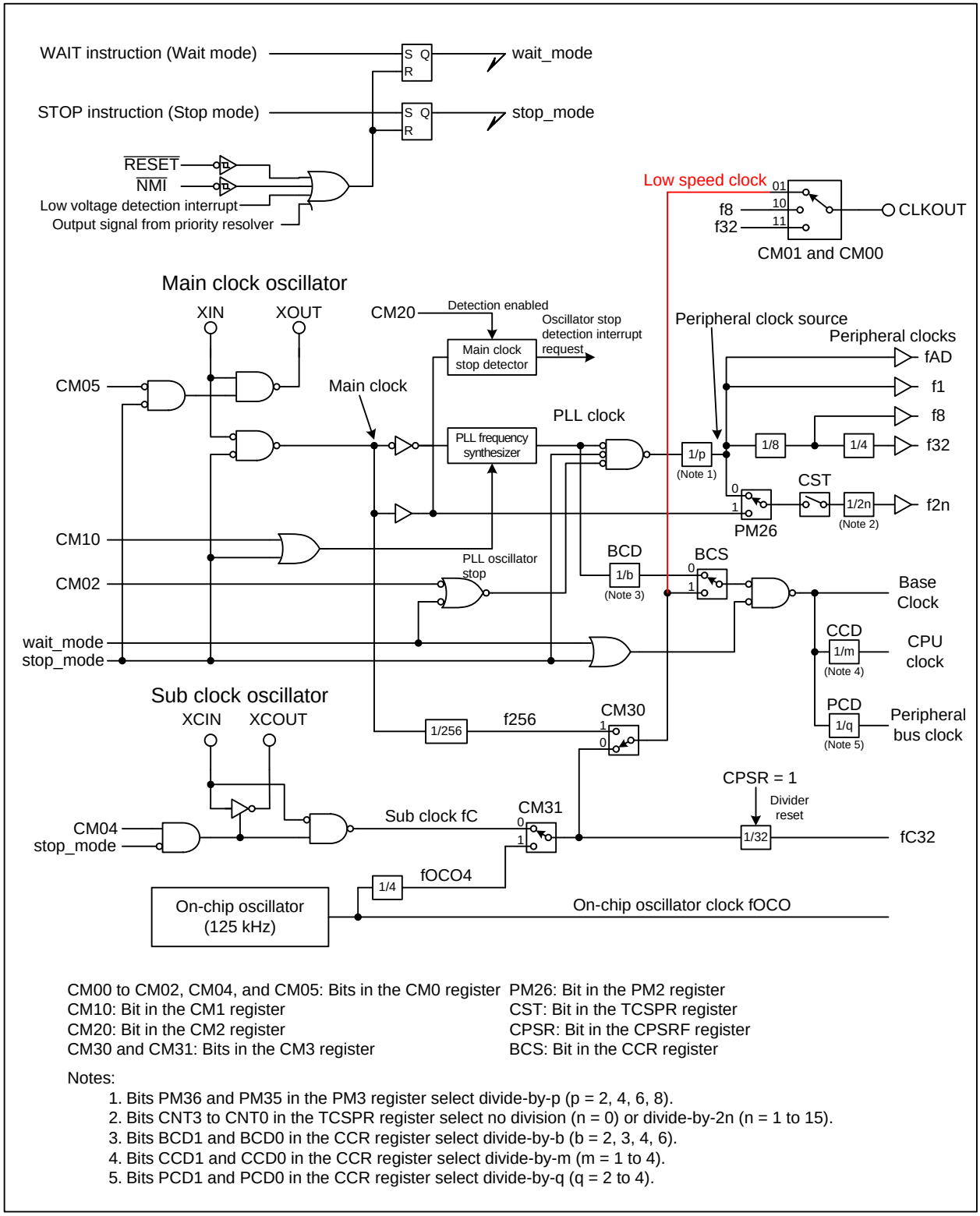
Note 2: “Before setting this bit to 1, set the VDEN bit to 1 (low voltage detecor enabled) first, and wait until the circuit is stablized.”

Note 3: “This bit is enabled when the VDEN bit is set to 1 (low voltage detecor enabled).”

•Page 88 of 624, description in 6.2.1 is corrected as follows:

“The low voltage detector starts operating stably after td(E-A) when the VDEN bit in the LVDC register is set to 1 (low voltage detector enabled).”

•Page 91 of 624, Figure 7.1 is corrected as follows:



CM00 to CM02, CM04, and CM05: Bits in the CM0 register
CM10: Bit in the CM1 register
CM20: Bit in the CM2 register
CM30 and CM31: Bits in the CM3 register
PM26: Bit in the PM2 register
CST: Bit in the TCSPR register
CPSR: Bit in the CPSRF register
BCS: Bit in the CCR register

Notes:
1. Bits PM36 and PM35 in the PM3 register select divide-by-p (p = 2, 4, 6, 8).
2. Bits CNT3 to CNT0 in the TCSPR register select no division (n = 0) or divide-by-2n (n = 1 to 15).
3. Bits BCD1 and BCD0 in the CCR register select divide-by-b (b = 2, 3, 4, 6).
4. Bits CCD1 and CCD0 in the CCR register select divide-by-m (m = 1 to 4).
5. Bits PCD1 and PCD0 in the CCR register select divide-by-q (q = 2 to 4).

Figure 7.1 Clock Generation Circuitry

•Page 92 of 624, descriptions of Notes 2 and 6 in Figure 7.2 are corrected as follows:
Note 2: "The divide ratios of the base clock and peripheral bus clock should not be changed simultaneously. Otherwise, the peripheral bus clock frequency may be over the operational maximum." ("To increase the base clock frequency, the divide ratio of the peripheral bus clock should be increased before reducing the divide ratio of base clock." is deleted)
Note 6: "To use these low speed clocks, select one of them by setting bits CM31 and CM30 in the CM3 register and then set the BCS bit to 1."

•Pages 93, 107, 114, and 117 of 624, descriptions "fC" in the function column of bits CM01 and CM00 in Figure 7.3, Section 7.6, Tables 7.3, 7.4, and 7.6 are corrected as follows:
Figure 7.3: "0 1 : Output a low speed clock"
Section 7.6: "Low speed clocks, f8, and f32 are available to be output from the CLKOUT pin."
Table 7.3: "Output a low speed clock"
Table 7.4: "When a low speed clock is selected"
Table 7.6: "When a low speed clock is selected"

•Page 93 of 624, description is added as Note 7 in Figure 7.3 as follows:
"Set this bit before activating the watchdog timer. When rewriting this bit while the watchdog timer is running, set it immediately after writing to the WDTS register."

•Page 94 of 624, description of bit name "PLL Clock Oscillator Stop Bit" in Figure 7.4 is corrected as follows:
"PLL Oscillator Stop Bit"

•Page 94 of 624, description is added as Note 4 in Figure 7.4 as follows:
"This bit becomes 1 when the main clock is stopped. Set it to 0 after the main clock oscillation is fully stabilized."

•Page 95 of 624, description is corrected to Note 1 in Figure 7.6 as follows:
"Rewrite this register after setting the PRC27 bit in the PRCR2 register to 1 (write enabled) and while the BCS bit in the CCR register is 0 (PLL clock)."

•Page 97 of 624, descriptions in Note 3 in Figure 7.9 are corrected as follows:
"CM05 bit in the CM0 register (main clock oscillator enabled/disabled)
CM10 bit in the CM1 register (PLL oscillator enabled/disabled)"

•Page 97 of 624, description is added as Note 6 in Figure 7.9 as follows:
"Disable all the peripheral functions that use f2n before rewriting this bit."

•Page 98 of 624, description is added to Note 1 in Figure 7.10 as follows:
"Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register. Disable all the peripheral functions that use fAD, f1, f8, f32, or f2n (when the clock source is the peripheral clock source) to rewrite this register."

•Page 102 of 624, descriptions for the SEO bit in Figure 7.15 are corrected as follows:

Bit Symbol	Bit Name	Function	RW
SEO	Self-Oscillating Setting Bit	0: PLL lock-in 1: Self-oscillating	RW

•Page 102 of 624, description is added as Note 1 in Figure 7.16 as follows:
"This register is reset after the SEO bit in the PLC1 register is set to 1 (self-oscillating). Stopping the main clock or PLL prevents the register from updating."

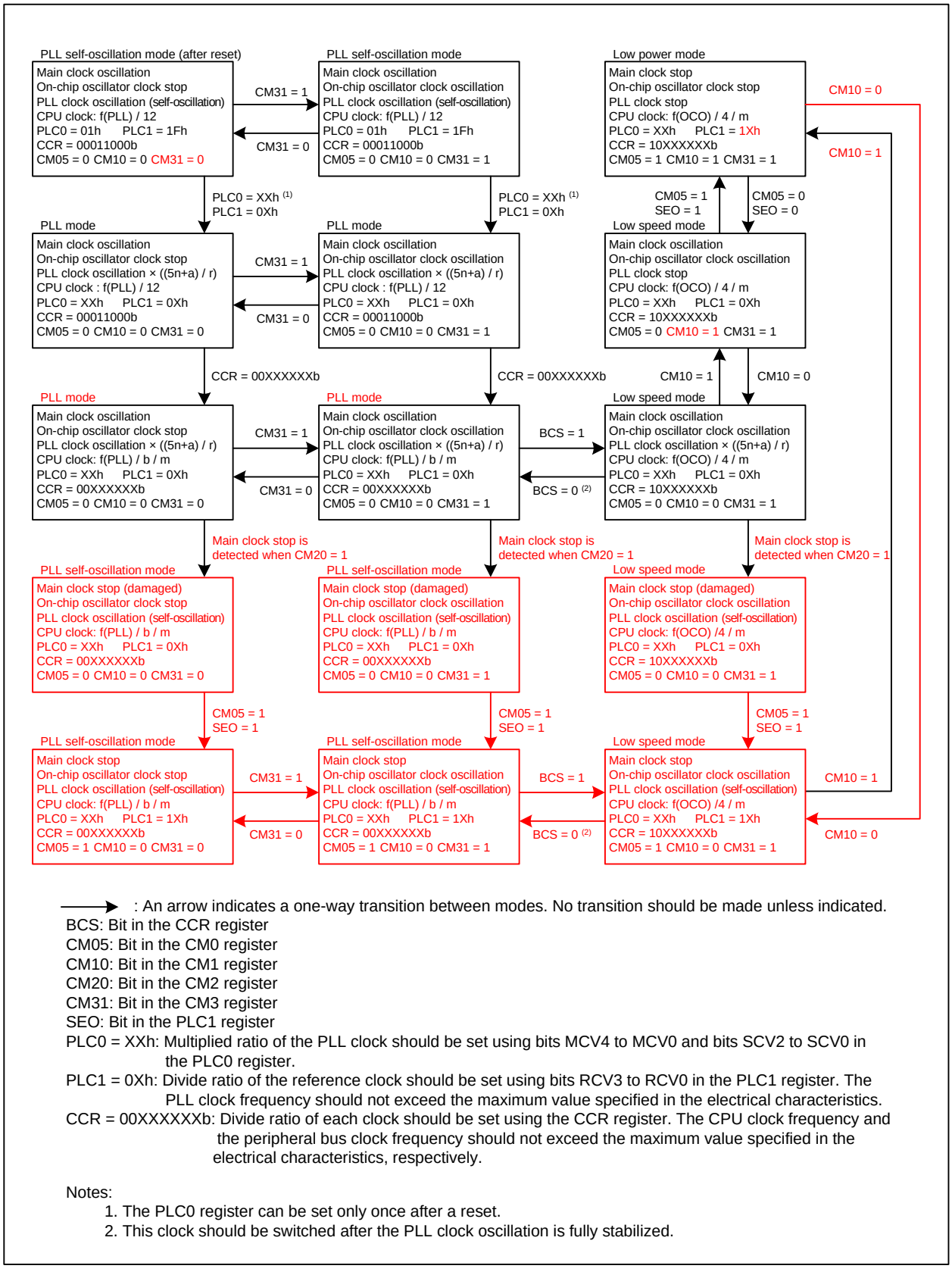
•Page 105 of 624, description in 7.2.1 is corrected as follows:

"When the main clock oscillator resumes running after an oscillator stop is detected, the PLL clock frequency may temporarily exceed the preset value before the PLL frequency synthesizer oscillation stabilizes. As soon as an oscillator stop is detected, the main clock oscillator should be stopped from resuming (set the CM05 bit in the CM0 register to 1) or the divide ratios of the base clock and peripheral clock source should be increased by a program. The respective divide ratio can be set by bits BCD1 and BCD0 in the CCR register and bits PM36 and PM35 in the PM3 register."

•Page 108 of 624, description of 7.7 is corrected as follows:

"Power control contains three modes: wait mode, stop mode, and normal operating mode.
The name "normal operating mode" is used restrictively in this chapter, and it indicates all other modes except wait mode and stop mode. Figure 7.16 shows a block diagram of the state transition in normal operating mode, stop mode, and wait mode."

•Page 111 of 624, descriptions of “CM31 = 1” in the first row and “CM10 = 0” in the second row in Figure 7.20 are corrected, and the flowchart is clarified as follows:



→ : An arrow indicates a one-way transition between modes. No transition should be made unless indicated.
BCS: Bit in the CCR register
CM05: Bit in the CM0 register
CM10: Bit in the CM1 register
CM20: Bit in the CM2 register
CM31: Bit in the CM3 register
SEO: Bit in the PLC1 register
PLC0 = XXh: Multiplied ratio of the PLL clock should be set using bits MCV4 to MCV0 and bits SCV2 to SCV0 in the PLC0 register.
PLC1 = 0Xh: Divide ratio of the reference clock should be set using bits RCV3 to RCV0 in the PLC1 register. The PLL clock frequency should not exceed the maximum value specified in the electrical characteristics.
CCR = 00XXXXXXb: Divide ratio of each clock should be set using the CCR register. The CPU clock frequency and the peripheral bus clock frequency should not exceed the maximum value specified in the electrical characteristics, respectively.

Notes:
1. The PLC0 register can be set only once after a reset.
2. This clock should be switched after the PLL clock oscillation is fully stabilized.

Figure 7.20 State Transition (when the on-chip oscillator clock is used)

(Note 3 “The CM05 bit is not set to 1 when an oscillator stop is detected through the oscillation stop detector.” is deleted from Figure 7.20)

•Page 114 of 624, descriptions of 7.7.2.4 are corrected as follows:

"Wait mode is exited by the hardware reset, an NMI, or peripheral interrupts assigned to software interrupt numbers from 0 to 63.

To exit wait mode by either the hardware reset or an NMI, without using peripheral interrupts, bits ILVL2 to ILVL0 for the peripheral interrupts should be set to 000b (interrupt disabled) before executing the WAIT instruction.

The CM02 bit setting in the CM0 register affects the peripheral interrupts. When the CM02 bit is set to 0 (peripheral clock source not stopped in wait mode), peripheral interrupts for software interrupt numbers from 0 to 63 can be used to exit wait mode. When this bit is set to 1 (peripheral clock source stopped in wait mode), peripheral functions operated using clocks (f1, f8, f32, f2n whose clock source is the peripheral clock source, and fAD) generated by the peripheral clock source stop operating. Therefore, the peripheral interrupts cannot be used to exit wait mode. However, peripheral functions operated using clocks which are independent from the peripheral clock source (fC32, external clock, and f2n whose clock source is the main clock) do not stop operating. Thus, interrupts generated by peripheral functions and assigned to software interrupt numbers from 0 to 63 can be used to exit wait mode.

The CPU clock used when exiting wait mode by the peripheral interrupts or an NMI is the same clock used when the WAIT instruction is executed.

Table 7.5 lists interrupts to be used to exit wait mode and usage conditions."

•Page 115 of 624, description of Note 1 of Table 7.5 is added as follows:

"INT6 to INT8 are available in the intelligent I/O interrupt only."

•Page 116 of 624, description of 7.7.3 is corrected as follows:

"In stop mode, all of the clocks, except for those that are protected, stop running."

•Page 117 of 624, description in line 1 of 7.7.3.3 is corrected as follows:

"Stop mode is exited by the hardware reset, an NMI, low voltage detection interrupt, or peripheral interrupts assigned to software interrupt numbers from 0 to 63."

•Page 117 of 624, description is added to Table 7.7 as follows:

External interrupt	INT6 to INT8 are available when intelligent I/O interrupt is used

•Page 143 of 624, Figure 10.8 is corrected as follows:

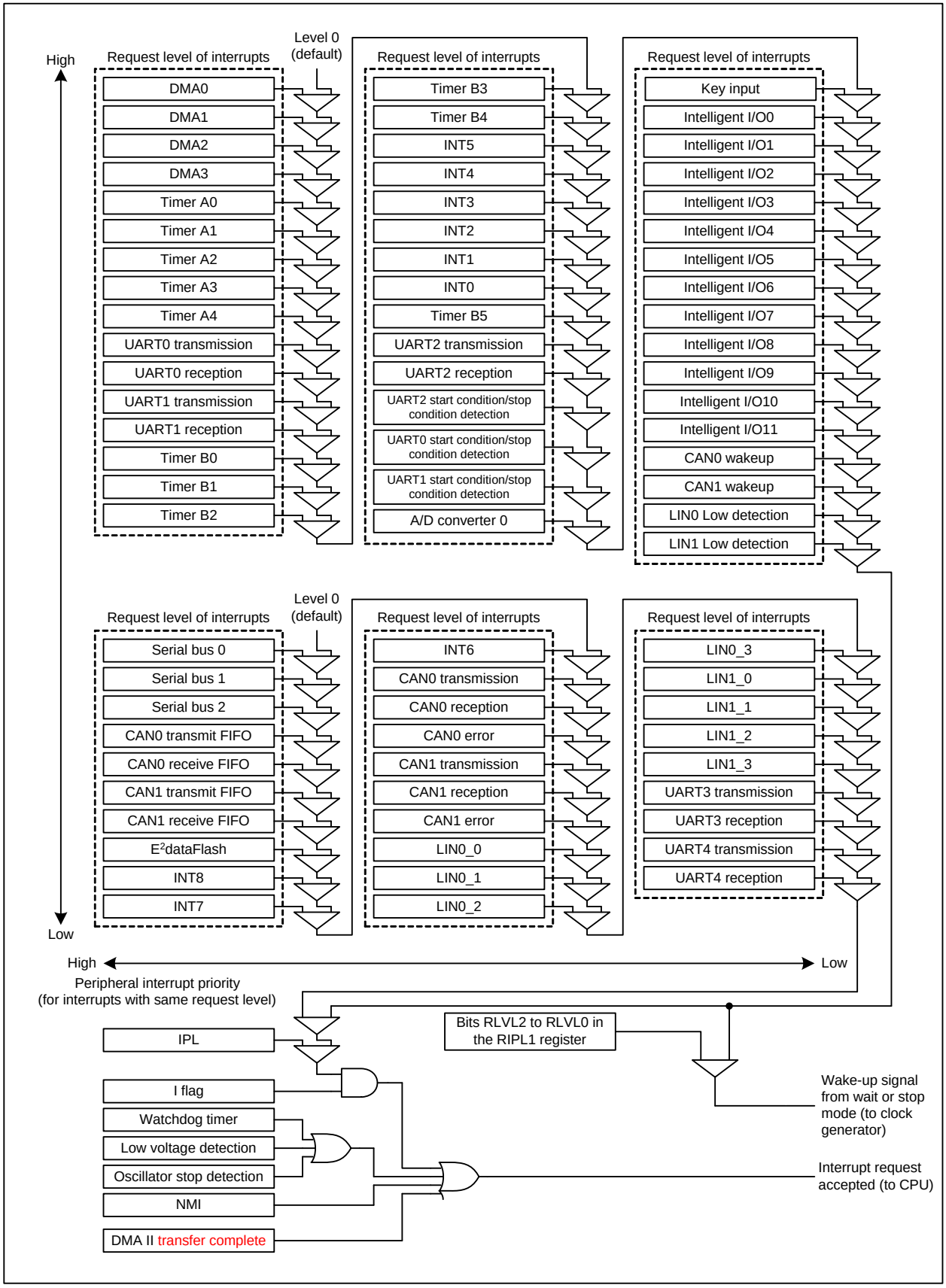


Figure 10.8 Priority Resolver

(Description “Bits RLVL2 to RLVL0 in the RIPL2 register” and associated signal lines are deleted from Figure 10.8)

•Page 150 of 624, descriptions for b0 and Note 3 in Figure 10.17 are corrected as follows:
b0: "No register bit; this bit is read as 1" ("should be written with 0 and" is deleted)
Note 3: "When this bit is function-assigned, it can be set to 0 only. It should not be set to 1. To set to 0, either the AND or BCLR instruction should be used; when the bit is not function-assigned, that is, reserved, it should be set to 0."

•Page 152 of 624, description of the third bullet point in 10.14.3 is corrected as follows:
"The interrupt input signals to pins $\overline{\text{INT6}}$ to $\overline{\text{INT8}}$ are also connected to bits INT6R to INT8R in registers IIO9IR to IIO11IR. Therefore, these input signals, when assigned to the intelligent I/O, can be used as a source for exiting wait mode or stop mode. Note that these signals are enabled only on the falling edge and not affected by the following bit settings: bits POL and LVS in the INTiIC register (i = 0 to 8), IFSR0i bit (i = 0 to 5) in the IFSR0 register, and the IFSR1j bit (j = i - 6; i = 6 to 8) in the IFSR1 register."

•Page 154 of 624, descriptions are added as Notes 1 and 2 to Figure 11.2 as follows:
Note 1: "When the on-chip oscillator clock is used as the count source, the read value may be undefined due to a change in the count value while being read."
Note2: "Set this bit before activating the watchdog timer."

•Page 155 of 624, description is added as Note 2 to Figure 11.3 as follows:
"Set these bits before activating the watchdog timer."

•Page 156 of 624, description is added as Note 3 to Figure 11.5 as follows:
"These bit settings are disabled when the WDTON bit is 1. The values set to these bits are reflected to registers WDK and PM2 when the WDTON bit is 0."

•Page 185 of 624, Figure 15.2 is corrected as follows:

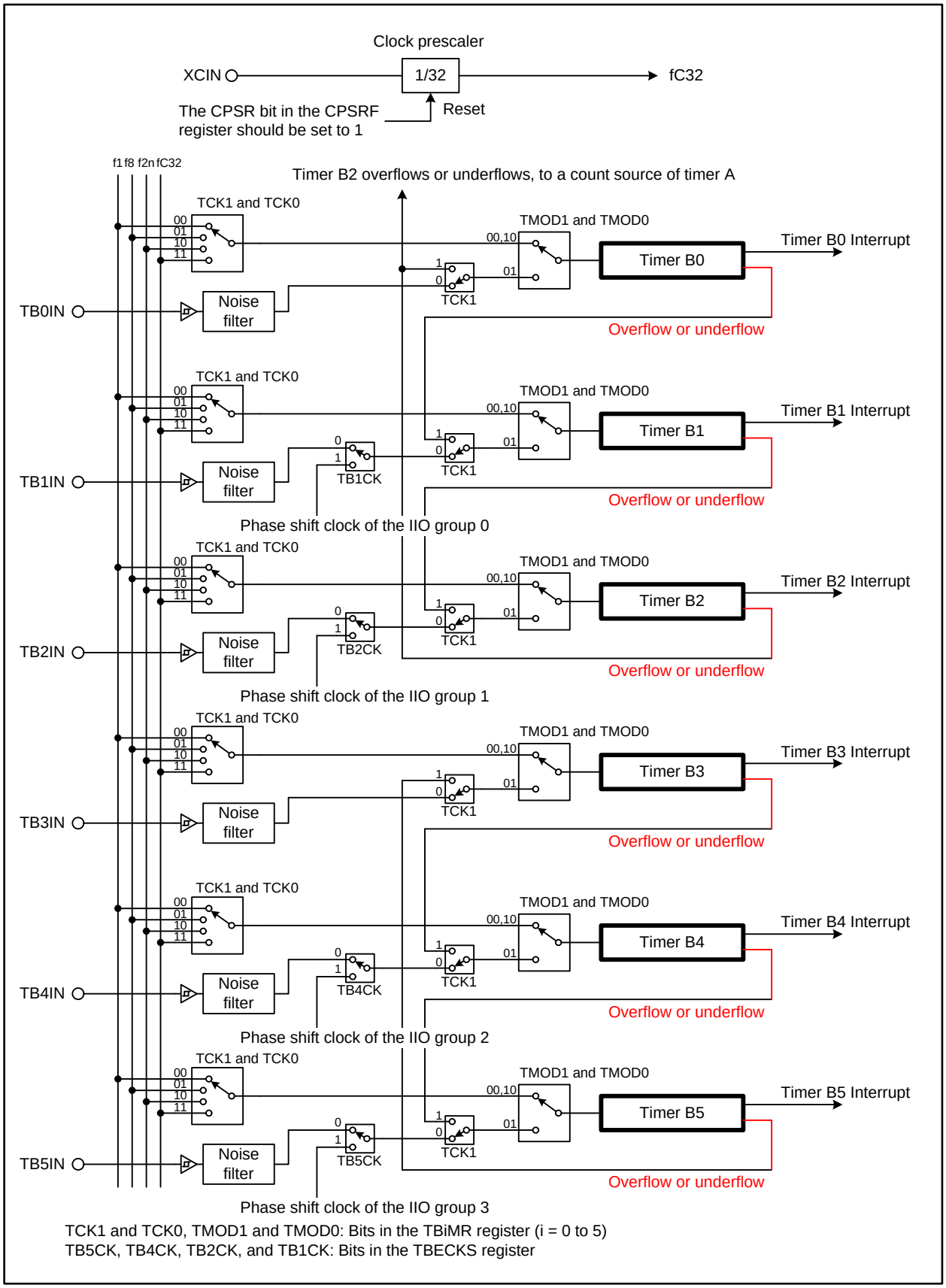


Figure 15.2 Timer B Configuration

•Pages 221 and 227 of 624, descriptions of functions of the INV13 bit in Figure 16.3 and the PWCON bit in Figure 16.9 are corrected as follows:

INV13: "0: Timer A1 reload control signal is 0
1: Timer A1 reload control signal is 1"

PWCON: "1: The underflow of timer B2 when the reload control signal for timer A1 is 0"

•Page 226 of 624, descriptions of functions of bits MR2 and MR3 in Figure 16.8 are corrected as follows:

MR2: "No register bit; should be written with 0 and read as undefined value"

MR3: "Disabled in the three-phase motor control timers. Should be written with 0 and read as undefined value"

•Page 234 of 624, description of case 1 in Figure 16.18 is corrected as follows:

"-INV01 = 0 and ICTB2 = 2h (timer B2 interrupt generated every 2nd time the timer B2 underflows), or INV01 = 1, INV00 = 1and ICTB2 = 1h (timer B2 interrupt generated every time the timer B2 underflows when the reload control signal for timer A1 is set to 1)"

•Page 236 of 624, descriptions in 16.6.1 are corrected as follows:

"When a low signal is applied to the NMI pin with the bit settings below, pins TA1OUT, TA2OUT, and TA4OUT become high-impedance: the PM24 bit in the PM2 register is 1 (NMI enabled), the SDE bit in the IOBC register is 1 (shutdown enabled), the INV02 bit in the INVC0 register is 1 (the three-phase motor control timers used), and the INV03 bit is 1 (the three-phase motor control timer output enabled)."

•Page 236 of 624, descriptions "overflows" and "overflow" in 16.6.2 are corrected as follows:

"Do not write to the TAi1 register (i = 1, 2, 4) in the timing that timer B2 underflows. Before writing to the TAi1 register, read the TB2 register to verify that sufficient time is left until timer B2 underflows. Then, immediately write to the TAi1 register so that no interrupt handler is performed during this write procedure. If the TB2 register indicates little time is left until the underflow, write to the TAi1 register after timer B2 underflows."

•Page 242 of 624, Note 1 "Bits CNT3 to CNT0 in the TCSPR register select a divide ratio from two options: no division (n = 0) or divide-by-2n (n = 1 to 15)." is deleted from Figure 17.5

•Page 246 of 624, description of function of the SWC bit in Figure 17.11 is corrected as follows:

"0: No wait-state/wait-state cleared
1: Hold the SCLi pin low after the eighth bit is received"

•Page 248 of 624, description of function of the SWC9 bit in Figure 17.13 is corrected as follows:

"0: No wait-state/wait-state cleared
1: Hold the SCLi pin low after the ninth bit is received"

•Pages 248 and 274 of 624, descriptions "STARREQ" in Note 3 of Figure 17.13 and 17.3.2 are corrected as follows:

Figure 17.13: "The STSPSEL bit should be set to 1 after setting the STAREQ, RSTAREQ, or STPREQ bit to 1."

17.3.2: "The start condition, restart condition, and stop condition are generated by bits STAREQ, RSTAREQ, and STPREQ in the UiSMR4 register (i = 0 to 2), respectively."

•Pages 266 and 267 of 624, descriptions "Transmit/receive clock" in Figures 17.27 and 17.28 are corrected as follows:

"CLKi"

•Page 276 of 624, descriptions in 17.3.4 are corrected as follows:

"Data transmission/reception in I^2C mode uses the transmit/receive clock as shown in Figure 17.30. The clock speed increase makes it difficult to secure the required time for ACK generation and data transmit procedure. The I^2C mode supports a function of wait-state insertion to secure this required time and a function of clock synchronization with a wait-state inserted by other devices.

The SWC bit in the UiSMR2 register (i = 0 to 2) is used to insert a wait-state for ACK generation. When the SWC bit is set to 1 (hold the SCLi pin low after the eighth bit is received), the SCLi pin is held low on the falling edge of the eighth bit of the SCLi. When the SWC bit is set to 0 (no wait-state/wait-state cleared), the SCLi line is released.

When the SWC2 bit in the UiSMR2 register is set to 1 (hold the SCLi pin low), the SCLi pin is forced low even during transmission or reception in progress. When the SWC2 bit is set to 0 (output the transmit/receive clock at the SCLi pin), the SCLi line is released to output the transmit/receive clock.

The SWC9 bit in the UiSMR4 register is used to insert a wait-state for checking received acknowledge bits. While the CKPH bit in the UiSMR3 register is set to 1 (clock delayed), when the SWC9 bit is set to 1 (hold the SCLi pin low after the ninth bit is received), the SCLi pin is held low on the falling edge of the ninth bit of the SCLi. When the SWC9 bit is set to 0 (no wait-state/wait-state cleared), the SCLi line is released."

•Pages 278, 282, and 283 of 624, descriptions "(external clock selected)" in line 1 of 17.3.8 and line 2 of 17.4.2.2 and "(internal clock selected)" in line 2 of 17.4.2.1 are corrected as follows:

"(external clock)" and "(internal clock)"

•Page 278 of 624, description "the SCLi pin is held low after the eighth bit of the SCLi is received)" in the third bullet point of 17.3.8 is corrected as follows:

"(hold the SCLi pin low after the eighth bit is received)"

•Page 284 of 624, description of the fourth dash in 17.5.2.1 is replaced as follows:

"- The TE bit in the UiC1 register is set to 1 (transmission enabled).

- The RE bit in the UiC1 register is set to 1 (reception enabled). This bit setting is not required in transmit operation only.

- The TI bit in the UiC1 register is set to 0 (data held in the UiTB register)."

•Page 307 of 624, description in the ninth bullet point of 18.3.2 is corrected as follows:

"The external trigger cannot be used in DMAC operating mode. When the DMAC is configured to transfer converted results, do not read the AD00 register by a program."

•Page 310 of 624, description "CRC_CCITT" in line 2 of Chapter 20 is corrected as follows:

"A generator polynomial of CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) generates a CRC."

•Pages 318 to 321 of 624, descriptions "Request from the $\overline{INT0}$ pin" in Figure 22.1, "Request from the $\overline{INT1}$ pin" in Figure 22.2, "Request from the $\overline{INT2}$ pin" in Figure 22.3, and "Request from the $\overline{INT3}$ pin" in Figure 22.4 are corrected as follows:

Figure 22.1: "Request from the $\overline{INT0}$ pin or the $\overline{INT1}$ pin"
Figure 22.2: "Request from the $\overline{INT0}$ pin or the $\overline{INT1}$ pin"
Figure 22.3: "Request from the $\overline{INT0}$ pin or the $\overline{INT1}$ pin"
Figure 22.4: "Request from the $\overline{INT0}$ pin or the $\overline{INT1}$ pin"

•Page 324 of 624, descriptions in the function column of the RST2 bit and Note 3 in Figure 22.7 are corrected as follows:

RST2: "1: A low signal input into the $\overline{INT0}$/$\overline{INT1}$ pin"

Note 3: "The base timer is reset by an input of low signal to the external interrupt pin selected for signals UD0Z (for groups 0 and 2) and UD1Z (for groups 1 and 3) by the IFS2 register."

•Page 332 of 624, description in the second bullet point of specification for reset conditions in Table 22.2 is corrected as follows:
"An input of low signal into the external interrupt pin as follows:
for groups 0 and 2: selected using bits IFS23 and IFS22 in the IFS2 register
for groups 1 and 3: selected using bits IFS27 and IFS26 in the IFS2 register"

•Page 333 of 624, description "Low signal input to the $\overline{\text{INTk}}$ pin" in Figure 22.17 is corrected as follows:
"Low signal input to the $\overline{\text{INT0/INT1}}$ pin"

•Page 354 of 624, bit name "SBUMS" in line 6 of 23. Serial Bus Interface is corrected as follows:
"SSUMS"

•Pages 355 and 357 of 624, pin name "SSiCK" and register name "SSiRDR" in Tables 23.1 and 23.2 are corrected as follows:
"SSCKi" and "SSiTDR"

•Page 357 of 624, descriptions "SSIi (I): Data input pin" and "SSOi (O): Data output pin" in the I/O pins row in Table 23.2 are corrected as follows:
"SSIi (I/O): Data I/O pin" and "SSOi (I/O): Data I/O pin"

•Page 361 of 624, description of function of bits BC2 to BC0 in Figure 23.5 is added as follows:

Bit Symbol	Bit Name	Function	RW
BC0	Bit Counter	Bits left during transmission/reception b2 b1 b0 0 0 0 : 8 bits left 0 0 1 : 1 bits left 0 1 0 : 2 bits left 0 1 1 : 3 bits left 1 0 0 : 4 bits left 1 0 1 : 5 bits left 1 1 0 : 6 bits left 1 1 1 : 7 bits left	RW
BC1			RW
BC2			RW

•Page 364 of 624, descriptions of functions of bits CE and ORER in Figure 23.8 are modified as follows:

Bit Symbol	Bit Name	Function	RW
CE	Conflict Error Flag (1, 3)	0: No conflict error 1: Conflict error occurred	RW
— (b1)	No register bit; should be written with 0 and read as 0		—
ORER	Overrun Error Flag (1, 3)	0: No overrun error 1: Overrun error occurred	RW

•Page 365 of 624, descriptions "input/output pin" and "Input/output pin" for the SCKS bit in Figure 23.9 are modified as follows:
"I/O pin"

•Pages 366, 375 to 379, 383, and 385 of 624, description “(overrun error)” for the ORER bit when it is 1 in Figure 23.11, 23.1.6.2 to 23.1.6.4, 23.1.7.2, and 23.1.7.3 is modified as follows:
“(overrun error occurred)”

•Page 387 of 624, description “(conflict error)” in line 5 of 23.1.7.4 is modified as follows:
“(conflict error occurred)”

•Page 389 of 624, descriptions “Break dominant” and “Break delimiter” in Table 24.1 are corrected as follows:
“Transmit break length” and “Transmit break delimiter length”

•Page 393 of 624, descriptions of functions of bits L3LD to L0LD in Figure 24.3 are corrected as follows:
L0LD: “The bit is read as the value of the LD bit in the LiST register of channel 0”
L1LD: “The bit is read as the value of the LD bit in the LiST register of channel 1”
L2LD: “The bit is read as the value of the LD bit in the LiST register of channel 2”
L3LD: “The bit is read as the value of the LD bit in the LiST register of channel 3”

•Page 393 of 624, description of Note 1 of Figure 24.3 is added as follows:
“No new interrupt is generated by the input signal low detection when any one of these bits is 1.”

•Page 395 of 624, description of Note 4 of Figure 24.7 is corrected as follows:
“The LD bit in the LiST register becomes 1 and an interrupt request is generated in the following cases:
- When the falling edge of the input signal is detected when this bit is set to 1.
- When this bit is set to 1 while the input signal is low.”

•Page 395 of 624, figure title of Figure 24.7 is corrected as follows:
“Registers L0MD0 and L1MD0”

•Page 398 of 624, descriptions of bit names of bits BLT3 to BLT0 and bits BDT1 and BDT0 in Figure 24.10 are corrected as follows:
Bits BLT3 to BLT0: “Transmit Break (Low) Length Setting Bit”
Bits BDT1 and BDT0: “Transmit Break Delimiter (High) Length Setting Bit”

•Page 401 of 624, figure title of Figure 24.13 is corrected as follows:
“Registers L0IDB and L1IDB”

•Pages 402 and 403 of 624, description of Note 1 of Figures 24.16 and 24.17 is corrected as follows:
“These bits do not become 0 automatically. Set them to 0 by a program. Writing 1 to these bits has no effect.”

•Page 402 of 624, description of Note 3 of Figure 24.16 is added as follows:
“When this bit is 1, no new interrupt request is generated in the following cases:
- When the LD bits of other channels become 1.
- When the conditions for the LD bit to become 1 are met in its channel.”

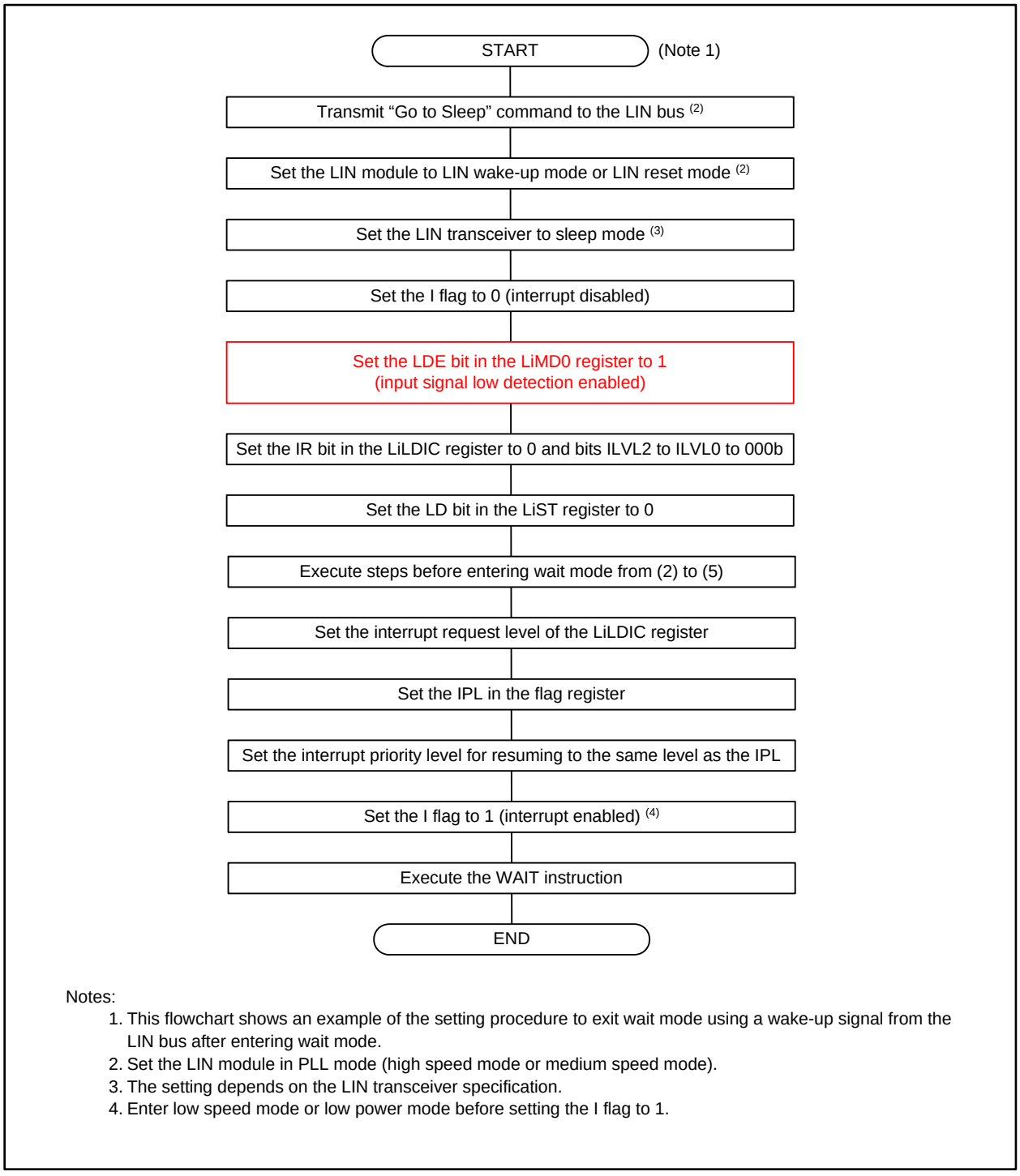
•Page 407 of 624, bit names in the fourth bullet point of Table 24.3 are changed as follows:
“Bits BLT3 to BLT0 for break low (13 to 28 Tbit);
Bits BDT1 and BDT0 for break delimiter (1 to 4 Tbit)”

•Page 408 of 624, descriptions for (4) for “LIN Module Processing” in Table 24.4 are changed as follows:
“Transmit Data 2, then the next interbyte space
Transmit Data 3, then the next interbyte space
(Repeat this process for the data length specified in bits RFDL3 to RFDL0 in the LiRFC register. Go to (6) if an error occurs.)”

•Page 409 of 624, descriptions for (4) for “LIN Module Processing” in Table 24.5 are changed as follows:
“Receive Data 2 due to start bit detection
Receive Data 3 due to start bit detection
(Repeat this process for the data length specified in bits RFDL3 to RFDL0 in the LiRFC register. Abort the reception and go to (5) if an error occurs. Checksum judgement is not performed in this case.)”

•Page 412 of 624, description “BPR0” in Note 1 of Table 24.6 is corrected as follows:
“BRP0”

•Page 419 of 624, Figure 24.30 is corrected as follows:



Notes:
1. This flowchart shows an example of the setting procedure to exit wait mode using a wake-up signal from the LIN bus after entering wait mode.
2. Set the LIN module in PLL mode (high speed mode or medium speed mode).
3. The setting depends on the LIN transceiver specification.
4. Enter low speed mode or low power mode before setting the I flag to 1.

Figure 24.30 Example of Setting Before Transmition to Wait Mode (i = 0, 1)

•Page 420 of 624, description for Detecting Condition for Input signal low detection in Table 24.8 is changed as follows:

"When the falling edge of input signal at the LINi_jIN pin is detected with the setting of the LDE bit in the LiMD0 register to 1 (Input signal low detection enabled), or when setting the LDE bit to 1 while the LINi_jIN pin is low."

•Page 421 of 624, value range "; j = 0 to 3" in the title of Table 24.9 is deleted.

•Page 423 of 624, description in lines 9 to 10 of 24.11 is added as follows:

"The respective interrupt request is output when the corresponding flag in the LiST register becomes 1 while the corresponding bit in the LiMD0 register is set to 1 (interrupt enabled). No new interrupt request is generated by the other sources if any of them is 1 since multiple interrupt sources are aggregated."

•Page 423 of 624, Figure 24.32 is corrected as follows:

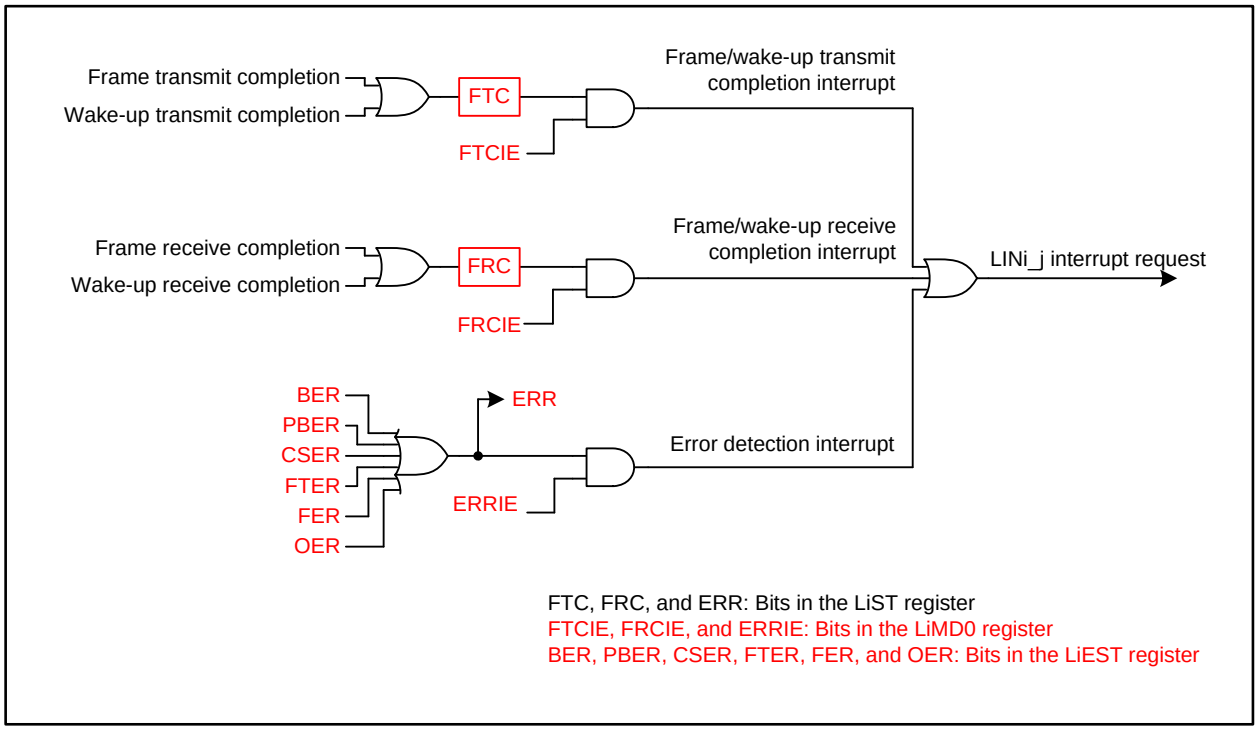


Figure 24.32 LINi_j Interrupt Block Diagram (i = 0, 1; j = 0 to 3)

•Page 424 of 624, Figure 24.33 is corrected as follows:

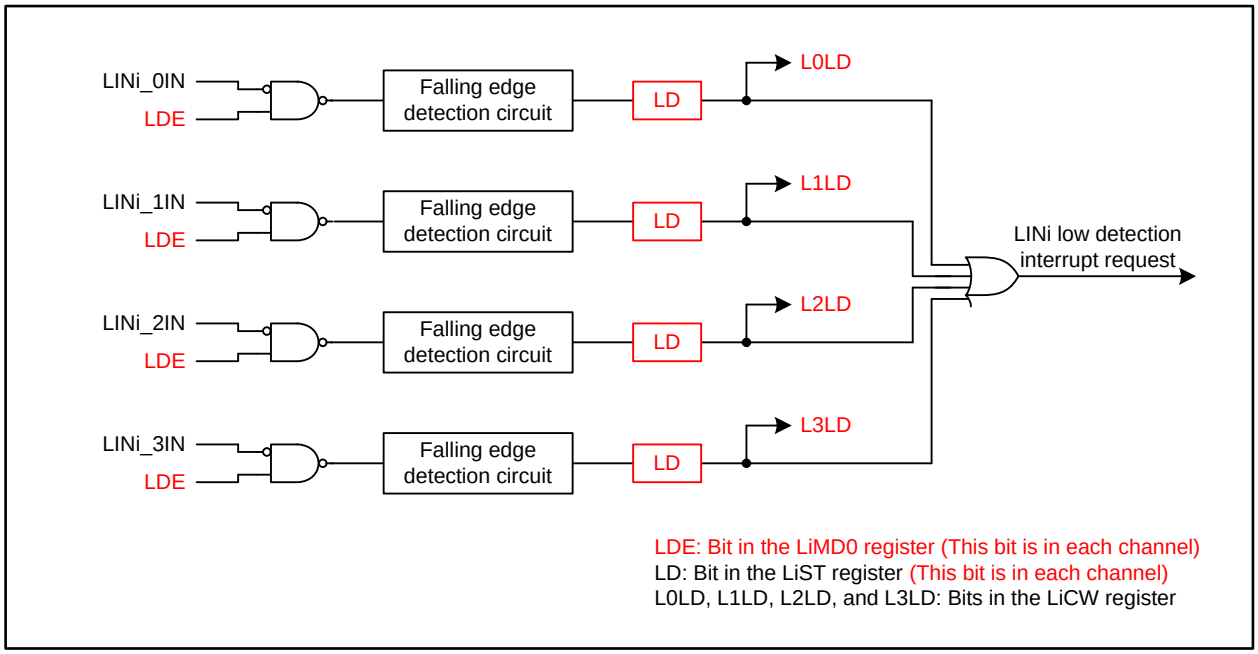


Figure 24.33 LINi Low Detection Interrupt Block Diagram (i = 0, 1)

•Page 448 of 624, description of Note 2 “Write 0 to the RFE bit simultaneously with the RFMLF bit.” in Figure 25.11 is corrected as follows:

“When setting the RFE bit to 0, set the RFMLF bit to 0 as well.”

•Page 470 of 624, description of Note 4 in Figure 25.28 is modified as follows:

“If more than one error condition is detected simultaneously, all corresponding bits are set to 1.”

•Page 480 of 624, description “CiSTR” in 25.2.4 is corrected as follows:

“Transmit mode: A CAN message is being transmitted. The CAN module may receive its own message simultaneously when self-test mode 0 (TSTM bit in the CiTCR register = 10b) or self-test mode 1 (TSTM bit = 11b) is selected.”

•Page 521 of 624, description of bit name “P14_1 to P14_3 Pull-Up Control Bit” in Figure 26.29 is corrected as follows:

PU40: “P14_1 and P14_3 Pull-Up Control Bit”

•Page 527 of 624, descriptions in Table 27.3 are corrected as follows:

Protection Type	Lock Bit Protection	ROM Code Protection	ID Code Protection
Operations to be protected	Erase, write	Read, write	Read, erase, write

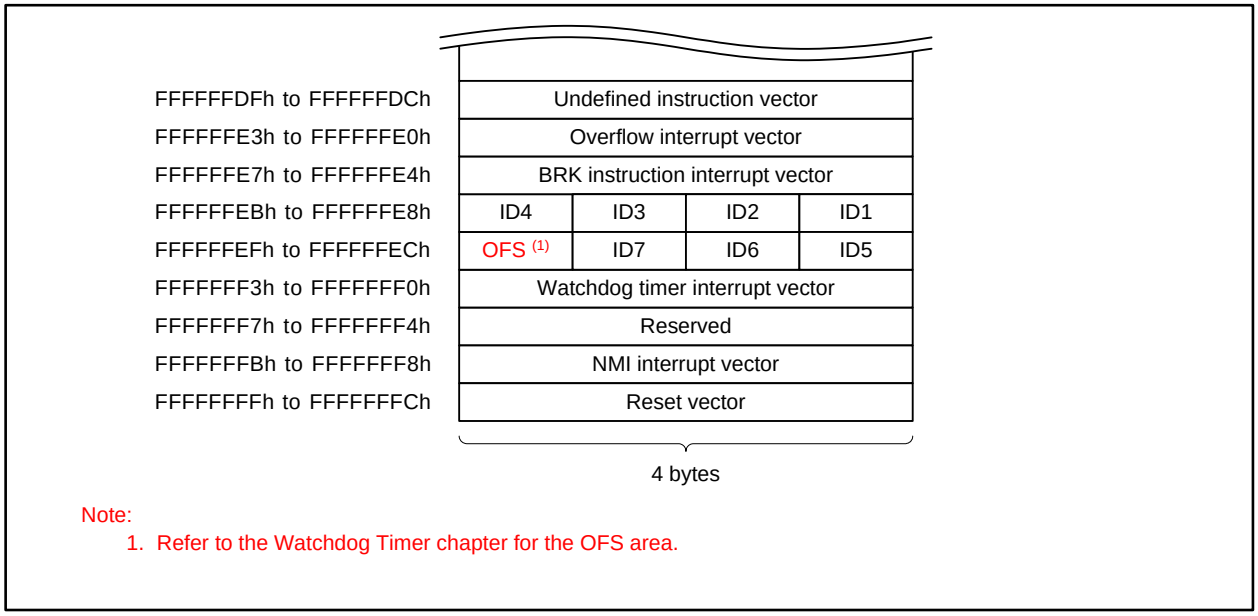
(Deleted description “erase” from the ROM Code Protection column)

Protection deactivated by	Setting the LBD bit in the FMR register to 1 (lock bit protection disabled). Or, by erasing the blocks whose lock bits are set to 0 to permanently deactivate the protection	Erasing all blocks whose protect bits are set to 0	Inputting a proper ID code to the serial programmer

(Deleted description “by using the serial programmer” from the ROM Code Protection column)

•Page 527 of 624, description “use the serial programmer to” in line 3 of 27.2.2 is deleted.

•Page 529 of 624, Figure 27.2 is corrected as follows:



Note:
1. Refer to the Watchdog Timer chapter for the OFS area.

Figure 27.2 Addresses for ID Code Stored

•Page 531 of 624, descriptions in Table 27.5 are corrected as follows:

Restriction on software command	None	• Do not execute either the program command or the block erase command for blocks where the rewrite control programs are written to • Do not execute the enter read status register mode command • Execute the enter read lock bit status mode command in RAM • Execute the enter read protect bit status mode command in RAM

Flash memory state detection by	• Reading the FMSR0 register by a program • Executing the enter read status register mode command to read data	• Reading the FMSR0 register by a program

•Page 531 of 624, description “Figure 27.11” in the last line below Table 27.5 is corrected as follows:
“Figure 27.12”

•Pages 538 and 540 of 624, descriptions in Figures 27.13 and 27.14 are corrected as follows:

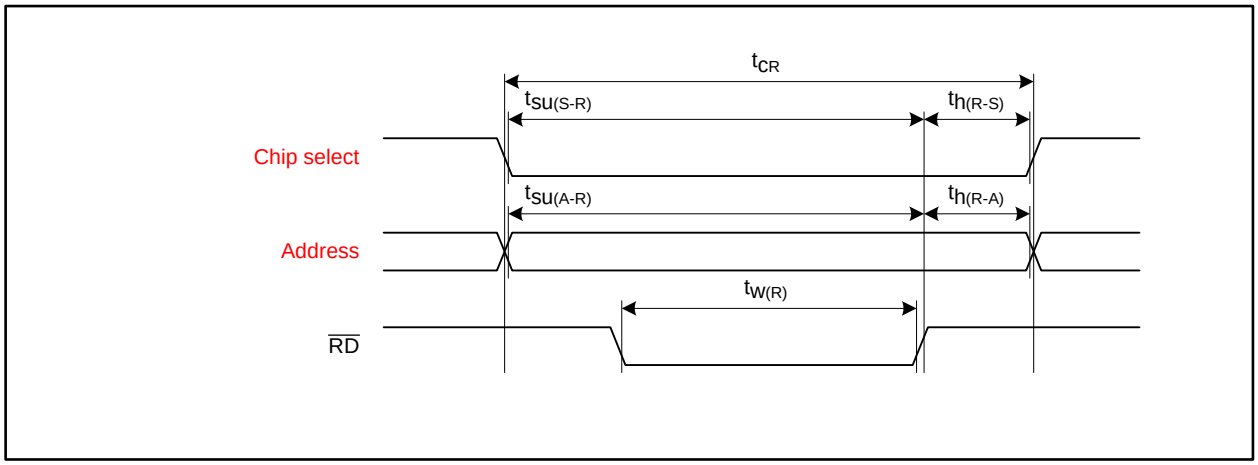


Figure 27.13 Read Timing

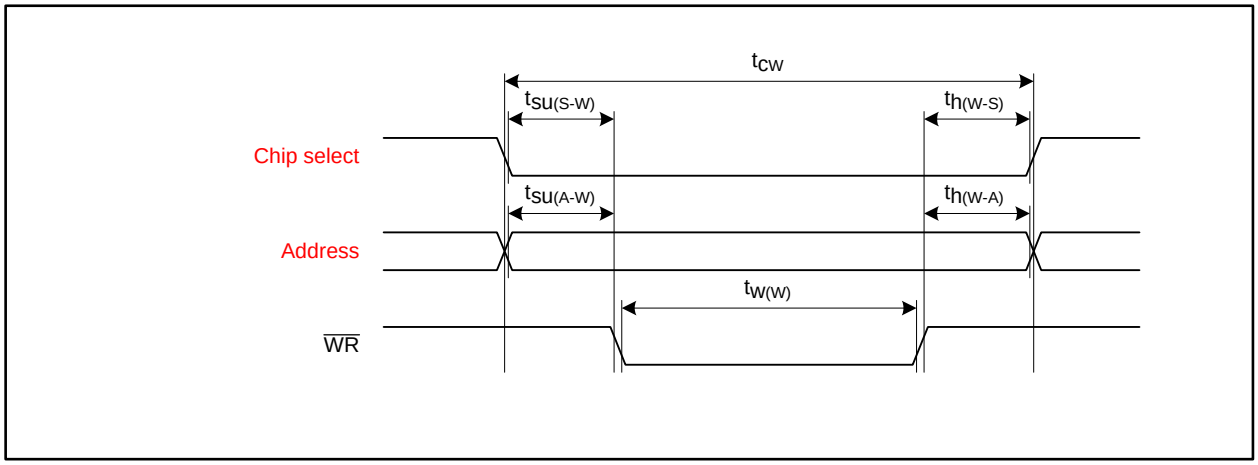


Figure 27.14 Write Timing

•Page 558 of 624, description of Note 3 in Figure 28.2 is corrected as follows:
"When this bit is set to 1, the module stops ongoing operations to enter its initial state. However, the registers are not initialized."

•Page 558 of 624, description "This mode setting prevents data from being overwritten if a program goes out of control." is deleted from Note 5 of Figure 28.2.

•Page 560 of 624, figure title "F2FI Register" for Figure 28.7 is corrected as follows:
"E2FI Register"

•Page 564 of 624, description for EERR bit setting in Figure 28.12 is corrected as follows:
"EERR bit in the E2FS0 register is 0?"

•Page 575 of 624, Table 29.13 is corrected as follows:

Symbol	Characteristics	Measurement condition	Value			Unit
			Min.	Typ.	Max.	
$f_{SO(PLL)}$	PLL clock self-oscillation frequency		35	57	80	MHz
$\|\Delta f_{LOCK}\|$	Lock detection (1)				2	%
$\|\Delta f_{UNLOCK}\|$	Unlock detection (1)		2			%
$t_{LOCK(PLL)}$	PLL lock time (2, 3)	$f_{(XRef)}$ = 4MHz			1	ms
$t_{jitter(p-p)}$	PLL jitter period (p-p)				2.0	ns
$f_{(OCO)}$	On-chip oscillator frequency		94	125	156	kHz

•Page 577 of 624, descriptions in Figure 29.5 are corrected as follows:

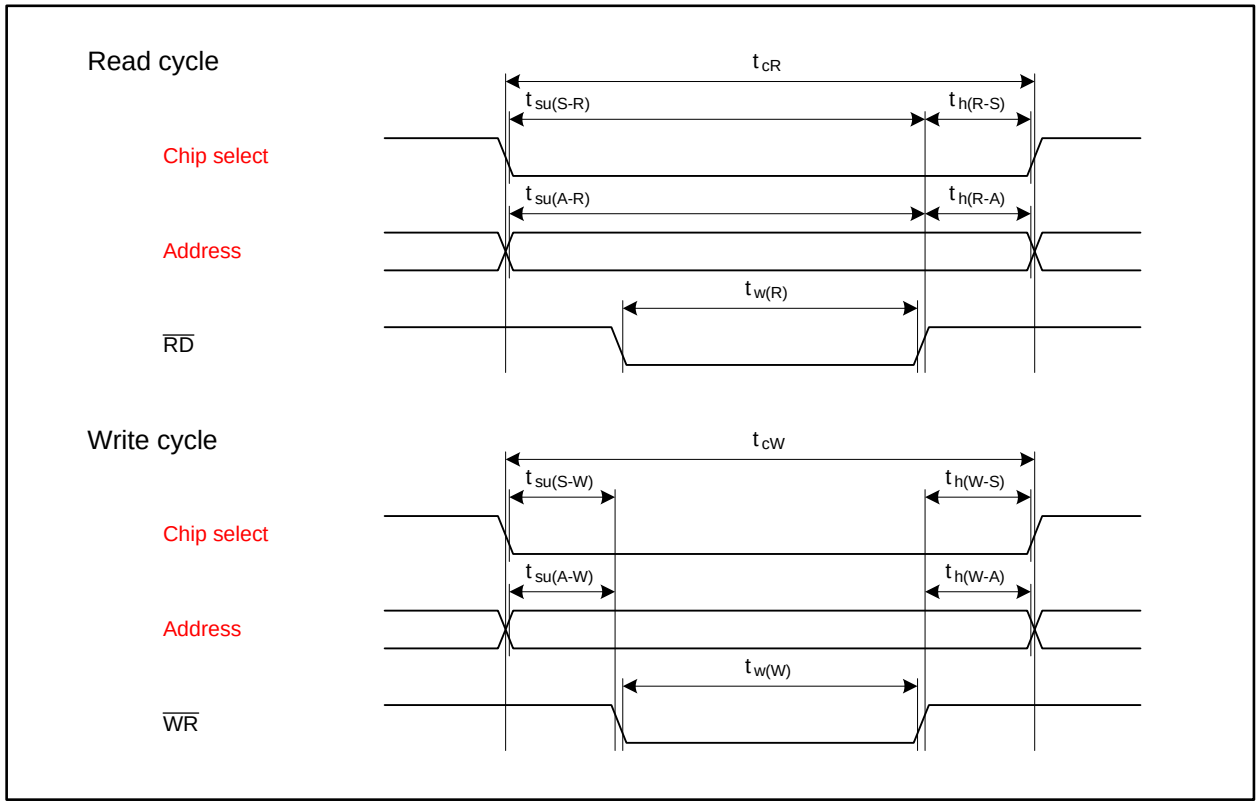


Figure 29.5 Flash Memory CPU Rewrite Mode Timing

•Page 607 of 624, description of the third bullet point in 30.4.3 is corrected as follows:

"The interrupt input signals to pins $\overline{INT6}$ to $\overline{INT8}$ are also connected to bits INT6R to INT8R in registers IIO9IR to IIO11IR. Therefore, these input signals, when assigned to the intelligent I/O, can be used as a source for exiting wait mode or stop mode. Note that these signals are enabled only on the falling edge and not affected by the following bit settings: bits POL and LVS in the INTiIC register (i = 0 to 8), IFSR0i bit (i = 0 to 5) in the IFSR0 register, and the IFSR1j bit (j = i - 6; i = 6 to 8) in the IFSR1 register."

•Page 612 of 624, descriptions in 30.7.1 are corrected as follows:

"When a low signal is applied to the NMI pin with the bit settings below, pins TA1OUT, TA2OUT, and TA4OUT become high-impedance: the PM24 bit in the PM2 register is 1 (NMI enabled), the SDE bit in the IOBC register is 1 (shutdown enabled), the INV02 bit in the INVC0 register is 1 (the three-phase motor control timers used), and the INV03 bit is 1 (the three-phase motor control timer output enabled)."

•Page 612 of 624, descriptions “overflows” and “overflow” in 30.7.2 are corrected as follows:
“Do not write to the TAi1 register (i = 1, 2, 4) in the timing that timer B2 underflows. Before writing to the TAi1 register, read the TB2 register to verify that sufficient time is left until timer B2 underflows. Then, immediately write to the TAi1 register so that no interrupt handler is performed during this write procedure. If the TB2 register indicates little time is left until the underflow, write to the TAi1 register after timer B2 underflows.”

•Page 613 of 624, description of the fourth dash in 30.8.2.1 is replaced as follows:
“- The TE bit in the UiC1 register is set to 1 (transmission enabled).
- The RE bit in the UiC1 register is set to 1 (reception enabled). This bit setting is not required in transmit operation only.
- The TI bit in the UiC1 register is set to 0 (data held in the UiTB register).”

•Page 615 of 624, description in the ninth bullet point of 30.9.2 is corrected as follows:
“The external trigger cannot be used in DMAC operating mode. When the DMAC is configured to transfer converted results, do not read the AD00 register by a program.”