RENESAS TECHNICAL UPDATE

TOYOSU FORESIA, 3-2-24, Toyosu, Koto-ku, Tokyo 135-0061, Japan
Renesas Electronics Corporation.

Product Category	MPU & MCU		Document No.	TN-RX*-A144A/E	Rev.	1.00
Title	Revision of the RX23T Group User's Manual due to changes in the section of Electrical Characteristics, additional functions and correction of description		Information Category	Technical Notification		
Applicable Product	RX23T Group	Lot No. All	Reference Document	RX23T Group User's Manual: Hardware Rev.1.00 (R01UH0520EJ0100)		

This is to inform you of erroneous description in the section of Electrical Characteristics in the RX23T Group, and of the following additional functions.

(1) High-speed on-chip oscillation function and general-purpose port added to the clock pins
(2) Expansion of the RAM capacity from 10 Kbytes to 12 Kbytes

Therefore, we revise the RX23T Group User's Manual: Hardware Rev.1.00 according the addition and correction of the functions mentioned above.

In addition, we also inform you of the corrections in other sections of the manual.

<Changes in the Section of Electrical Characteristics>

•Page 1051, Table 35.3 DC Characteristics (1) is corrected as follows:

Before correction

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Schmitt trigger input voltage	RIIC input pin (except for SMBus, 5 V tolerant)	V_{IH}	VCC × 0.7	—	5.8	V	
	Ports B1 and B2 (5 V tolerant)		VCC × 0.8	—	5.8		
	Ports 00 to 02, 10,11 Ports 22 to 24 Ports 30 to 33 Ports 70 to 76 Ports 91 to 94 Ports A2 to A5 Ports B0, B3 to B7 Ports D3 to D7 Port E2 Port RES#		VCC × 0.8	—	VCC + 0.3		
	Ports 40 to 47		AVCC0 × 0.8	—	AVCC0 + 0.3		
	RIIC input pin (except for SMBus)	V_{IL}	–0.3	—	VCC × 0.3		
	Other than RIIC input pin		–0.3	—	VCC × 0.2		
	RIIC input pin (except for SMBus)	ΔV_T	VCC × 0.05	—	—		
	Other than RIIC input pin		VCC × 0.1	—	—		
Input level voltage (except for Schmitt trigger input pins)	MD	V_{IH}	VCC × 0.9	—	VCC + 0.3	V	
	EXTAL (external clock input)		VCC × 0.8	—	VCC + 0.3		
	RIIC input pin (SMBus)		2.1	—	VCC + 0.3		
	MD	V_{IL}	–0.3	—	VCC × 0.1		
	EXTAL (external clock input)		–0.3	—	VCC × 0.2		
	RIIC input pin (SMBus)		–0.3	—	0.8		

Corrections

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Schmitt trigger input voltage	RIIC input pin (except for SMBus, 5 V tolerant)	V_{IH}	VCC × 0.7	—	5.8	V	
	Ports B1 and B2 (5 V tolerant)		VCC × 0.8	—	5.8		
	Ports 00 to 02, 10, 11 Ports 22 to 24 Ports 30 to 33, 36, 37 Ports 70 to 76 Ports 91 to 94 Ports A2 to A5 Ports B0, B3 to B7 Ports D3 to D7 Port E2 Port RES#		VCC × 0.8	—	VCC + 0.3		
	Ports 40 to 47		AVCC0 × 0.8	—	AVCC0 + 0.3		
	RIIC input pin (except for SMBus)	V_{IL}	–0.3	—	VCC × 0.3		
	Ports 40 to 47		–0.3	—	AVCC0 × 0.2		
	Other than RIIC input pin or ports 40 to 47		–0.3	—	VCC × 0.2		
	RIIC input pin (except for SMBus)	ΔV_T	VCC × 0.05	—	—		
	Ports 40 to 47		AVCC0 × 0.1	—	—		
	Other than RIIC input pin or ports 40 to 47		VCC × 0.1	—	—		
Input level voltage (except for Schmitt trigger input pins)	MD	V_{IH}	VCC × 0.9	—	VCC + 0.3	V	
	EXTAL (external clock input)		VCC × 0.8	—	VCC + 0.3		
	RIIC input pin (SMBus)		2.1	—	VCC + 0.3		
	MD	V_{IL}	–0.3	—	VCC × 0.1		
	EXTAL (external clock input)		–0.3	—	VCC × 0.2		
	RIIC input pin (SMBus)		–0.3	—	0.8		

•Page 1066, Table 35.14 Operating Frequency Value (High-Speed Operating Mode) is corrected as follows:

Before correction

Item		Symbol	min.	typ.	max.	Unit
Maximum operating frequency	System clock (ICLK)	f_{max}	1	—	40	MHz
	FlashIF clock (FCLK)*1, *2		1	—	32	
	Peripheral module clock (PCLKA)		1	—	40	
	Peripheral module clock (PCLKB)		1	—	40	
	Peripheral module clock (PCLKD)		1	—	40	

Corrections

Item		Symbol	min.	typ.	max.	Unit
Maximum operating frequency	System clock (ICLK)	f_{max}	—	—	40	MHz
	FlashIF clock (FCLK)*1, *2		—	—	32	
	Peripheral module clock (PCLKA)		—	—	40	
	Peripheral module clock (PCLKB)		—	—	40	
	Peripheral module clock (PCLKD)		—	—	40	

•Page 1066, Table 35.15 Operating Frequency Value (Middle-Speed Operating Mode) is corrected as follows:

Before correction

Item		Symbol	min.	typ.	max.	Unit
Maximum operating frequency	System clock (ICLK)	f_{max}	1	—	12	MHz
	FlashIF clock (FCLK)*1, *2		1	—	12	
	Peripheral module clock (PCLKA)		1	—	12	
	Peripheral module clock (PCLKB)		1	—	12	
	Peripheral module clock (PCLKD)		1	—	12	

Corrections

Item		Symbol	min.	typ.	max.	Unit
Maximum operating frequency	System clock (ICLK)	f_{max}	—	—	12	MHz
	FlashIF clock (FCLK)*1, *2		—	—	12	
	Peripheral module clock (PCLKA)		—	—	12	
	Peripheral module clock (PCLKB)		—	—	12	
	Peripheral module clock (PCLKD)		—	—	12	

•Page 1067, Table 35.16 Clock Timing is corrected as follows:

Before correction

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
EXTAL external clock input cycle time	t_{Xcyc}	50	—	—	ns	Figure 35.20
EXTAL external clock input high pulse width	t_{XH}	20	—	—	ns	
EXTAL external clock input low pulse width	t_{XL}	20	—	—	ns	
EXTAL external clock rise time	t_{Xr}	—	—	5	ns	
EXTAL external clock fall time	t_{Xf}	—	—	5	ns	
EXTAL external clock input wait time*1	t_{EXWT}	0.5	—	—	μs	
Main clock oscillator oscillation frequency*2	f_{MAIN}	1	—	20	MHz	
Main clock oscillation stabilization time (crystal)*2	$t_{MAINOSC}$	—	3	—	ms	Figure 35.21
Main clock oscillation stabilization time (ceramic resonator)*2	$t_{MAINOSC}$	—	50	—	μs	
LOCO clock oscillation frequency	f_{LOCO}	3.44	4.0	4.56	MHz	
LOCO clock oscillation stabilization time	t_{LOCO}	—	—	0.5	μs	Figure 35.22
IWDT-dedicated clock oscillation frequency	f_{ILOCO}	12.75	15	17.25	kHz	
IWDT-dedicated clock oscillation stabilization time	t_{ILOCO}	—	—	50	μs	Figure 35.23
PLL circuit oscillation frequency	f_{PLL}	24	—	40	MHz	
PLL clock oscillation stabilization time	t_{PLL}	—	—	50	μs	Figure 35.24
PLL free-running oscillation frequency	f_{PLLFR}	—	8	—	MHz	

Corrections

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
EXTAL external clock input cycle time	t_{Xcyc}	50	—	—	ns	Figure 35.20
EXTAL external clock input high pulse width	t_{XH}	20	—	—	ns	
EXTAL external clock input low pulse width	t_{XL}	20	—	—	ns	
EXTAL external clock rise time	t_{Xr}	—	—	5	ns	
EXTAL external clock fall time	t_{Xf}	—	—	5	ns	
EXTAL external clock input wait time*1	t_{EXWT}	0.5	—	—	μs	
Main clock oscillator oscillation frequency*2	f_{MAIN}	1	—	20	MHz	
Main clock oscillation stabilization time (crystal)*2	$t_{MAINOSC}$	—	3	—	ms	Figure 35.21
Main clock oscillation stabilization time (ceramic resonator)*2	$t_{MAINOSC}$	—	50	—	μs	
LOCO clock oscillation frequency	f_{LOCO}	3.44	4.0	4.56	MHz	
LOCO clock oscillation stabilization time	t_{LOCO}	—	—	0.5	μs	Figure 35.22
IWDT-dedicated clock oscillation frequency	f_{ILOCO}	12.75	15	17.25	kHz	
IWDT-dedicated clock oscillation stabilization time	t_{ILOCO}	—	—	50	μs	Figure 35.23
HOCO clock oscillation frequency	f_{HOCO}	31.52	32	32.48	MHz	T_a = –40 to +85°C
		31.68	32	32.32		T_a = –20 to +85°C
		31.36	32	32.64		T_a = –40 to +105°C
HOCO clock oscillation stabilization time	t_{HOCO}	—	—	30	μs	Figure 35.25
PLL circuit oscillation frequency	f_{PLL}	24	—	40	MHz	
PLL clock oscillation stabilization time	t_{PLL}	—	—	50	μs	Figure 35.26
PLL free-running oscillation frequency	f_{PLLFR}	—	8	—	MHz	

•Page 1067, Figure 35.20 XTAL External Clock Input Timing is corrected as follows:

Before correction

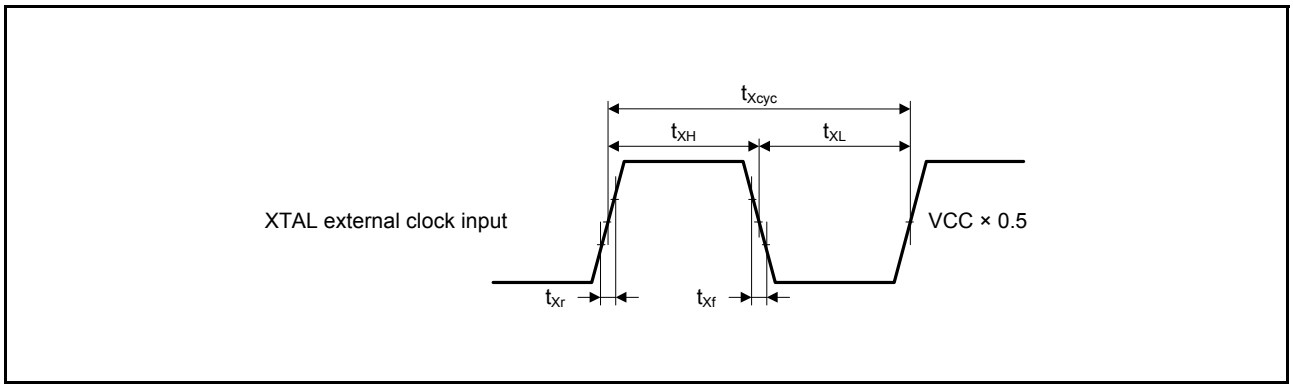


Figure 35.20 XTAL External Clock Input Timing

Corrections

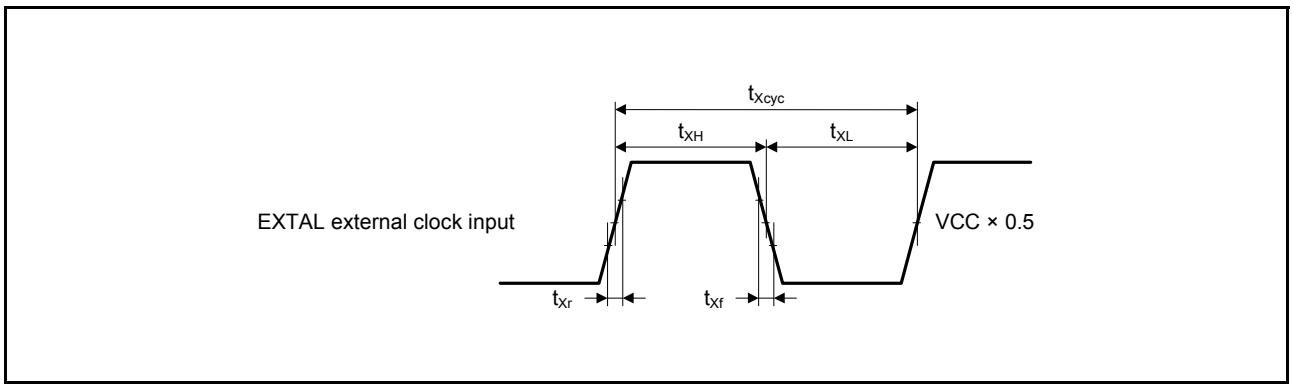


Figure 35.20 EXTAL External Clock Input Timing

•Page 1068, Figure 35.24 HOCO Clock Oscillation Start Timing (After Reset is Canceled by Setting OFS1.HOCOEN Bit to 0) is added as follows:

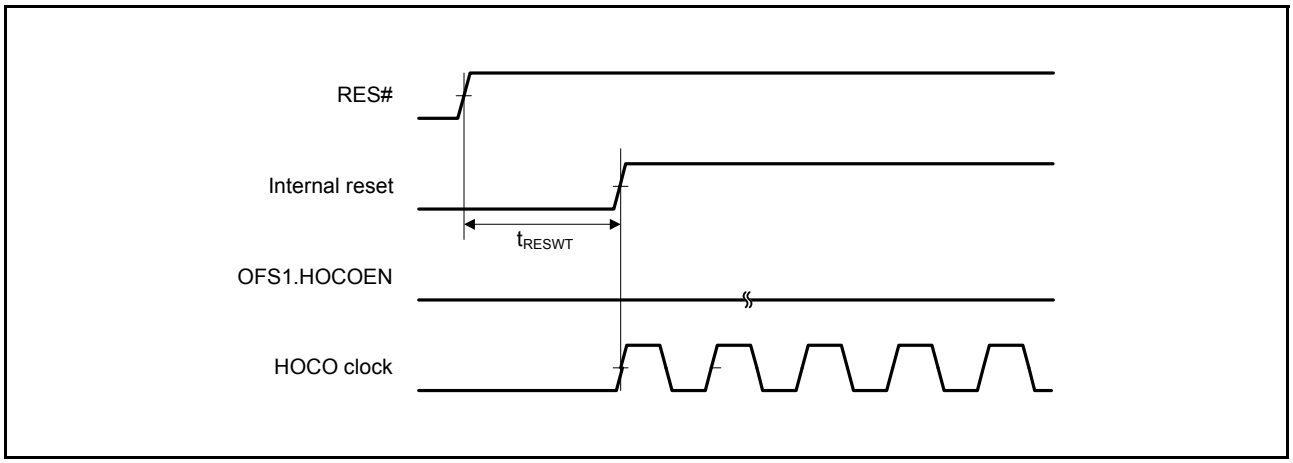


Figure 35.24 HOCO Clock Oscillation Start Timing (After Reset is Canceled by Setting OFS1.HOCOEN Bit to 0)

•Page 1068, Figure 35.25 HOCO Clock Oscillation Start Timing (Oscillation is Started by Setting HOCOCR.HCSTP Bit) is added as follows:

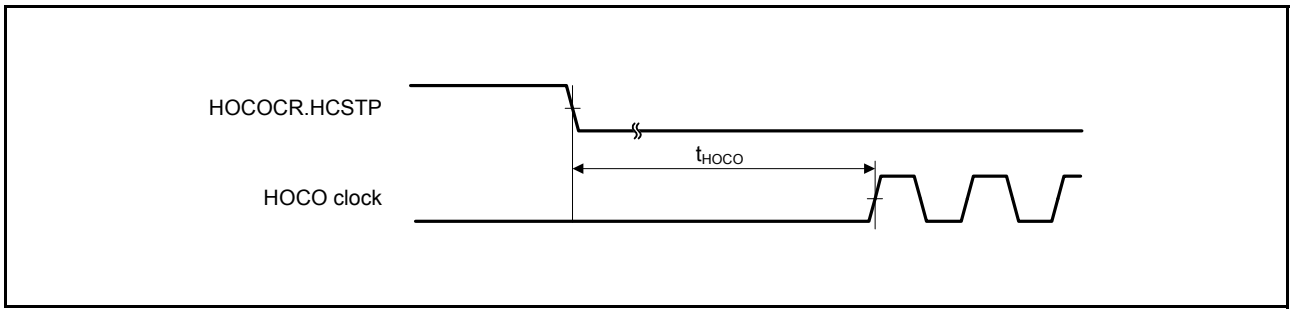


Figure 35.25 HOCO Clock Oscillation Start Timing (Oscillation is Started by Setting HOCOCR.HCSTP Bit)

•Page 1087, Table 35.32 Comparator Characteristics is corrected as follows:

Before correction

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Response time	t_{cr}	—	—	200	ns	VOD = 100 mV PCLKB = 40 MHz
	t_{cf}	—	—	200	ns	CMPCTL.ODFS = 0

Corrections

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Response time	t_{cr}	—	—	200	ns	VOD = 100 mV CMPCTL.CDFS = 0
	t_{cf}	—	—	200	ns	

•Page 1088, Table 35.33 D/A Conversion Characteristics is corrected as follows:

Before correction

Item	Min.	Typ.	Max.	Unit	Test Conditions
Resolution	—	—	8	Bit	
Conversion time	—	—	3.0	µs	
Absolute accuracy	—	±1.0	±3.0	LSB	

Corrections

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Resolution	—	—	—	8	Bit	
Conversion time	t_{DCONV}	—	—	3.0	µs	
Absolute accuracy	—	—	±1.0	±3.0	LSB	

•Page 1090, Table 35.35 Power-On Reset Circuit and Voltage Detection Circuit Characteristics (2) is corrected as follows:

Before correction

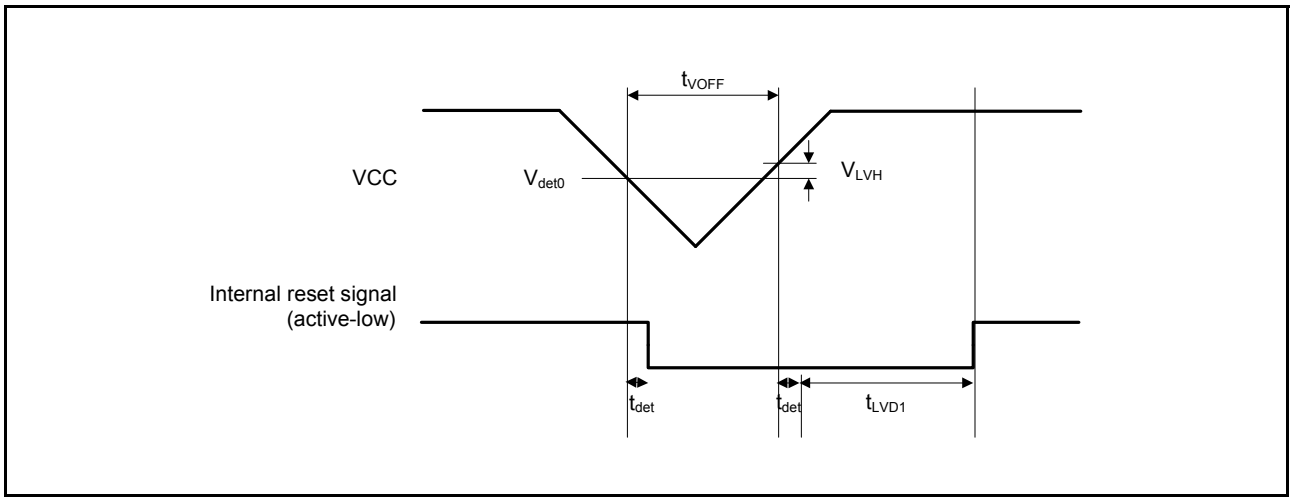
Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Wait time after power-on reset cancellation	t_{POR}	—	28.4	—	ms	Figure 35.49
Wait time after voltage monitoring 1 reset cancellation	t_{LVD1}	—	568	—	µs	Figure 35.51
Wait time after voltage monitoring 2 reset cancellation	t_{LVD2}	—	100	—	µs	Figure 35.52
Response delay time	t_{det}	—	—	350	µs	Figure 35.48
Minimum VCC down time*1	t_{VOFF}	350	—	—	µs	Figure 35.48, VCC = 1.0 V or above
Power-on reset enable time	$t_{W(POR)}$	1	—	—	ms	Figure 35.49, VCC = below 1.0 V
LVD operation stabilization time (after LVD is enabled)	$Td_{(E-A)}$	—	—	300	µs	Figure 35.51, Figure 35.52
Hysteresis width (LVD1 and LVD2)	V_{LVH}	—	70	—	mV	Vdet1_0 to 4 selected
		—	60	—		Vdet1_5 to 8, LVD2 selected

Corrections

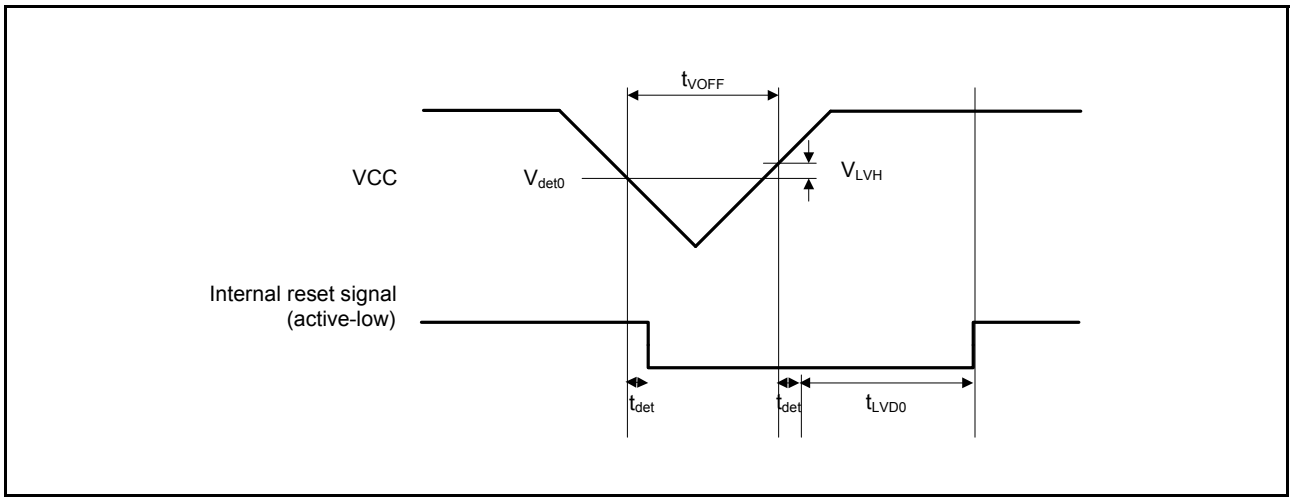
Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Wait time after power-on reset cancellation	t_{POR}	—	28.4	—	ms	Figure 35.51
Wait time after voltage monitoring 0 reset cancellation	t_{LVD0}	—	568	—	µs	Figure 35.52
Wait time after voltage monitoring 1 reset cancellation	t_{LVD1}	—	100	—	µs	Figure 35.53
Wait time after voltage monitoring 2 reset cancellation	t_{LVD2}	—	100	—	µs	Figure 35.54
Response delay time	t_{det}	—	—	350	µs	Figure 35.50
Minimum VCC down time*1	t_{VOFF}	350	—	—	µs	Figure 35.50, VCC = 1.0 V or above
Power-on reset enable time	$t_{W(POR)}$	1	—	—	ms	Figure 35.51, VCC = below 1.0 V
LVD operation stabilization time (after LVD is enabled)	$Td_{(E-A)}$	—	—	300	µs	Figure 35.53, Figure 35.54
Hysteresis width (LVD1 and LVD2)	V_{LVH}	—	70	—	mV	Vdet1_0 to 4 selected
		—	60	—		Vdet1_5 to 8, LVD2 selected

•Page 1091, Figure 35.50 Voltage Detection Circuit Timing (Vdet0) is corrected as follows:

Before correction



Corrections



<Description of the Additional Functions and Relevant Corrections>

•Page 34, the description in “Features” is corrected as follows:

Before correction

■ **Up to 48 pins for general I/O ports**

Corrections

■ **Up to 50 pins for general I/O ports**

•Page 34, the description of “On-chip SRAM, no wait states” in “Features” is corrected as follows:

Before correction

- 10 Kbytes of SRAM

Corrections

- 12 Kbytes of SRAM

•Page 34, the description of “Clock functions” in “Features” is corrected as follows:

Before correction

- On-chip low-speed oscillators, dedicated on-chip oscillator for the IWDT

Corrections

- On-chip low-speed oscillator, on-chip high-speed oscillator, dedicated on-chip oscillator for the IWDT

•Page 35, Table 1.1 Outline of Specifications (1/3) is corrected as follows:

Before correction

Classification	Module/Function	Description
Memory	RAM	· Capacity: 10 Kbytes
Clock	Clock generation circuit	· Main clock oscillator, low-speed on-chip oscillator, PLL frequency synthesizer, and IWDT-dedicated on-chip oscillator

Corrections

Classification	Module/Function	Description
Memory	RAM	· Capacity: 12 Kbytes
Clock	Clock generation circuit	· Main clock oscillator, low-speed and high-speed on-chip oscillator, PLL frequency synthesizer, and IWDT-dedicated on-chip oscillator

•Page 36, Table 1.1 Outline of Specifications (2/3) is corrected as follows:

Before correction

Classification	Module/Function	Description
I/O ports	General I/O ports	· I/O: 48/38/35 · Pull-up resistors: 48/38/35 · Open-drain outputs: 40/30/27

Corrections

Classification	Module/Function	Description
I/O ports	General I/O ports	· I/O: 50/40/37 · Pull-up resistors: 50/40/37 · Open-drain outputs: 42/32/29

•Page 39, Table 1.3 List of Products: D Version (T_a = –40 to +85°C) is corrected as follows:

Before correction

Group	Part No.	Package	ROM Capacity	RAM Capacity	Operating Frequency	Operating Temperature
RX23T	R5F523T5ADFL	PLQP0048KB-A	128 Kbytes	10 Kbytes	40MHz	-40 to + 85°C
	R5F523T5ADFD	PLQP0052JA-A				
	R5F523T5ADFM	PLQP0064KB-A				
	R5F523T3ADFL	PLQP0048KB-A	64 Kbytes			
	R5F523T3ADFD	PLQP0052JA-A				
	R5F523T3ADFM	PLQP0064KB-A				

Corrections

Group	Part No.	Package	ROM Capacity	RAM Capacity	Operating Frequency	Operating Temperature
RX23T	R5F523T5ADFL	PLQP0048KB-B	128 Kbytes	12 Kbytes	40MHz	-40 to + 85°C
	R5F523T5ADFD	PLQP0052JA-B				
	R5F523T5ADFM	PLQP0064KB-C				
	R5F523T3ADFL	PLQP0048KB-B	64 Kbytes			
	R5F523T3ADFD	PLQP0052JA-B				
	R5F523T3ADFM	PLQP0064KB-C				

•Page 39, Table 1.4 List of Products: G Version (T_a = –40 to +105°C) is corrected as follows:

Before correction

Group	Part No.	Package	ROM Capacity	RAM Capacity	Operating Frequency	Operating Temperature
RX23T	R5F523T5AGFL	PLQP0048KB-A	128 Kbytes	10 Kbytes	40MHz	-40 to +105°C
	R5F523T5AGFD	PLQP0052JA-A				
	R5F523T5AGFM	PLQP0064KB-A				
	R5F523T3AGFL	PLQP0048KB-A	64 Kbytes			
	R5F523T3AGFD	PLQP0052JA-A				
	R5F523T3AGFM	PLQP0064KB-A				

Corrections

Group	Part No.	Package	ROM Capacity	RAM Capacity	Operating Frequency	Operating Temperature
RX23T	R5F523T5AGFL	PLQP0048KB-B	128 Kbytes	12 Kbytes	40MHz	-40 to +105°C
	R5F523T5AGFD	PLQP0052JA-B				
	R5F523T5AGFM	PLQP0064KB-C				
	R5F523T3AGFL	PLQP0048KB-B	64 Kbytes			
	R5F523T3AGFD	PLQP0052JA-B				
	R5F523T3AGFM	PLQP0064KB-C				

•Page 40, Figure 1.1 How to Read the Product Part Number is corrected as follows:

Before correction

ROM/RAM capacity
5: 128 Kbytes/10 Kbytes
3: 64 Kbytes/10 Kbytes

Corrections

ROM/RAM capacity
5: 128 Kbytes/12 Kbytes
3: 64 Kbytes/12 Kbytes

•Page 43, Table 1.5 Pin Functions (2/2) is corrected as follows:

Before correction

Classifications	Pin Name	I/O	Description
I/O ports	P30 to P33	I/O	4-bit input/output pins.

Corrections

Classifications	Pin Name	I/O	Description
I/O ports	P30 to P33, P36, P37	I/O	6-bit input/output pins.

•Page 44, 45, and 46, Figure 1.3 Pin Assignments of the 64-Pin LFQFP, Figure 1.4 Pin Assignments of the 52-Pin LQFP, and Figure 1.5 Pin Assignments of the 48-Pin LFQFP are corrected as follows:

Before correction

XTAL
EXTAL

Corrections

P37/XTAL
P36/EXTAL

•Page 47, Table 1.6 List of Pins and Pin Functions (64-Pin LFQFP) (1/2) is corrected as follows:

Before correction

Pin No.	Power Supply, Clock, System Control	I/O Port	Timers (MTU, TMR, POE, CAC)	Communications (SCIg, RSPI, RIIC)	Others
7	XTAL	(blank)	(blank)	(blank)	(blank)
9	EXTAL	(blank)	(blank)	(blank)	(blank)

Corrections

Pin No.	Power Supply, Clock, System Control	I/O Port	Timers (MTU, TMR, POE, CAC)	Communications (SCIg, RSPI, RIIC)	Others
7	XTAL	P37	(blank)	(blank)	(blank)
9	EXTAL	P36	(blank)	(blank)	(blank)

•Page 49, Table 1.7 List of Pins and Pin Functions (52-Pin LQFP) is corrected as follows:

Before correction

Pin No.	Power Supply, Clock, System Control	I/O Port	Timers (MTU, TMR, POE, CAC)	Communications (SCIg, RSPI, RIIC)	Others
5	XTAL	(blank)	(blank)	(blank)	(blank)
7	EXTAL	(blank)	(blank)	(blank)	(blank)

Corrections

Pin No.	Power Supply, Clock, System Control	I/O Port	Timers (MTU, TMR, POE, CAC)	Communications (SCIg, RSPI, RIIC)	Others
5	XTAL	P37	(blank)	(blank)	(blank)
7	EXTAL	P36	(blank)	(blank)	(blank)

•Page 50, Table 1.8 List of Pins and Pin Functions (48-Pin LFQFP) is corrected as follows:

Before correction

Pin No.	Power Supply, Clock, System Control	I/O Port	Timers (MTU, TMR, POE, CAC)	Communications (SCIg, RSPI, RIIC)	Others
4	XTAL	(blank)	(blank)	(blank)	(blank)
6	EXTAL	(blank)	(blank)	(blank)	(blank)

Corrections

Pin No.	Power Supply, Clock, System Control	I/O Port	Timers (MTU, TMR, POE, CAC)	Communications (SCIg, RSPI, RIIC)	Others
4	XTAL	P37	(blank)	(blank)	(blank)
6	EXTAL	P36	(blank)	(blank)	(blank)

•Page 84, Figure 4.1 Memory Map in Each Operating Mode is corrected as follows:

Before correction

RAM (bytes)

Capacity	**Address**
10 Kbytes	0000 0000h to 0000 27FFh

Corrections

RAM (bytes)

Capacity	**Address**
12 Kbytes	0000 0000h to 0000 27FFh 0000 4000h to 0000 4A7Fh

•Page 87, the following registers are added to Table 5.1 List of I/O Registers (Address Order) (1/16):

Address	**Module Symbol**	**Register Name**	**Register Symbol**	**Number of Bits**	**Access Size**	**Number of Access Cycles** ICLK ≥PCLK	**Reference Section**
0008 0036h	SYSTEM	High-Speed On-Chip Oscillator Control Register	HOCOCR	8	8	3ICLK	section 9.
0008 00A5h	SYSTEM	High-Speed On-Chip Oscillator Wait Control Register	HOCOWTCR	8	8	3ICLK	section 9.

•Page 97, the following registers are added to Table 5.1 List of I/O Registers (Address Order) (11/16):

Address	**Module Symbol**	**Register Name**	**Register Symbol**	**Number of Bits**	**Access Size**	**Number of Access Cycles** ICLK ≥PCLK	**Reference Section**
0008 C087h	PORT3	Open Drain Control Register 1	ODR1	8	8, 16	2 or 3 PCLKB	section 18

•Page 118, “Option Function Select Register 1 (OFS1)” is corrected as follows:
The HOCOEN bit is added in b8.

Before correction

Address(es): FFFF FF88h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	The value set by the user*1															

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	LVDAS	VDSEL[1:0]	
Value after reset:	The value set by the user*1															

Bit	Symbol	Bit Name	Description	R/W
b1, b0	VDSEL[1:0]	Voltage Detection 0 Level Select	b1 b0 0 0: 3.84 V is selected 1 0: 2.51 V is selected Settings other than above are prohibited when the voltage detection 0 circuit is used.	R
b2	LVDAS	Voltage Detection 0 Circuit Start	0: Voltage monitoring 0 reset is enabled after a reset 1: Voltage monitoring 0 reset is disabled after a reset	R
b31 to b3	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R

Corrections

Address(es): FFFF FF88h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	The value set by the user*1															

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	HOCO EN	—	—	—	—	—	LVDAS	VDSEL[1:0]	
Value after reset:	The value set by the user*1															

Bit	Symbol	Bit Name	Description	R/W
b1, b0	VDSEL[1:0]	Voltage Detection 0 Level Select	b1 b0 0 0: 3.84 V is selected 1 0: 2.51 V is selected Settings other than above are prohibited when the voltage detection 0 circuit is used.	R
b2	LVDAS	Voltage Detection 0 Circuit Start	0: Voltage monitoring 0 reset is enabled after a reset 1: Voltage monitoring 0 reset is disabled after a reset	R
b7 to b3	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R
b8	HOCOEN	HOCO Oscillation Enable	0: HOCO oscillation is enabled after a reset 1: HOCO oscillation is disabled after a reset	R
b31 to b9	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R

(Additional description)

HOCOEN Bit (HOCO Oscillation Enable)

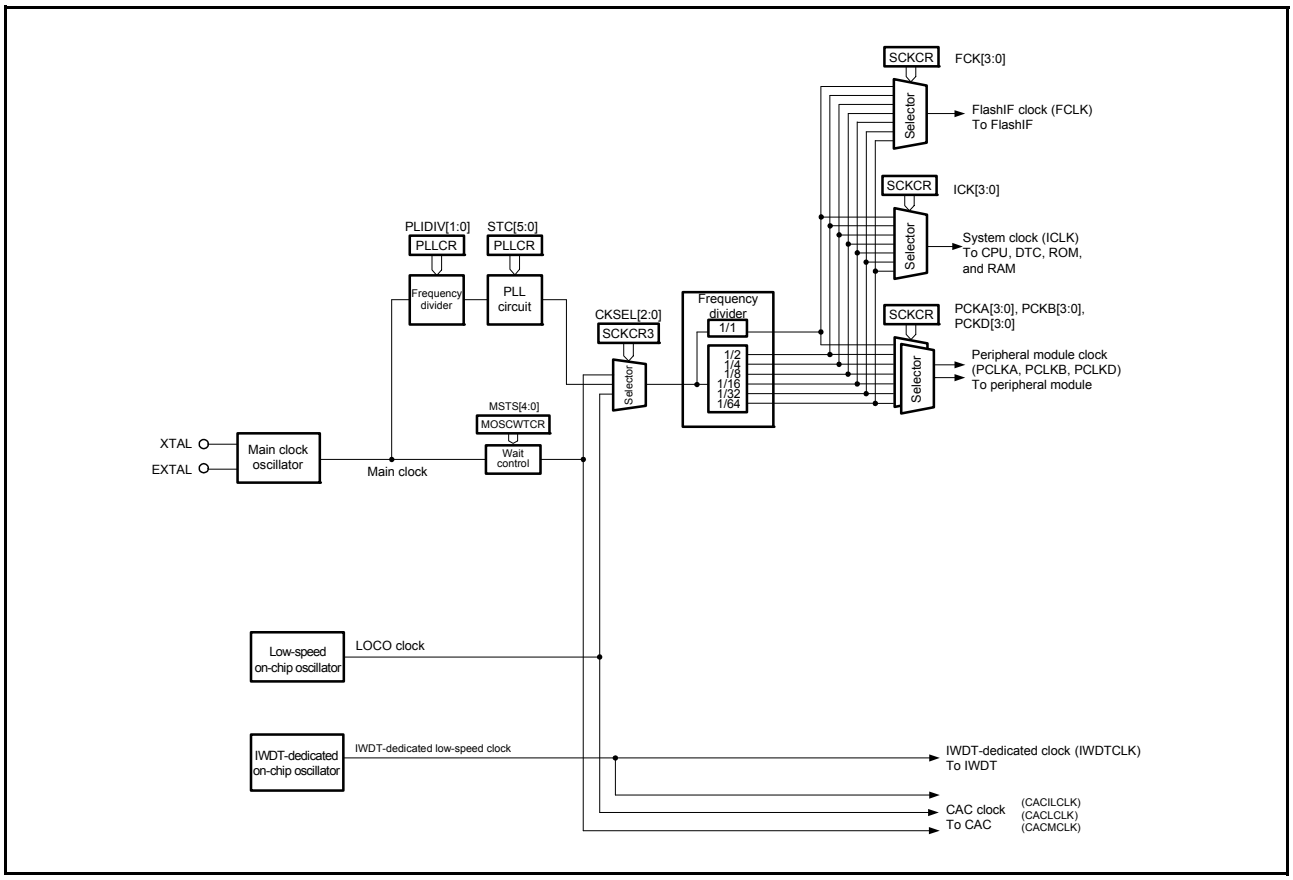
This bit selects whether the HOCO oscillation is effective or not after a reset.
Setting the HOCOEN bit to 0 allows the HOCO oscillation to be started before the CPU starts operation, and therefore reduces the wait time for oscillation stabilization.
Note that even if the HOCOEN bit is set to 0, the system clock source is not switched to HOCO. The system clock source is switched to HOCO only by modifying the clock source select bits (SCKCR3.CKSEL[2:0]) from the CPU.
Also, when the HOCOEN bit is set to 0, the HOCO oscillation stabilization time (tHOCO) is secured by hardware, so the clock with the accuracy of the HOCO oscillation frequency (fHOCO) shown in Electrical Characteristics is supplied after release from the CPU reset state.

•Page 137, the following item is added to Table 9.1 Specifications of Clock Generation Circuit:

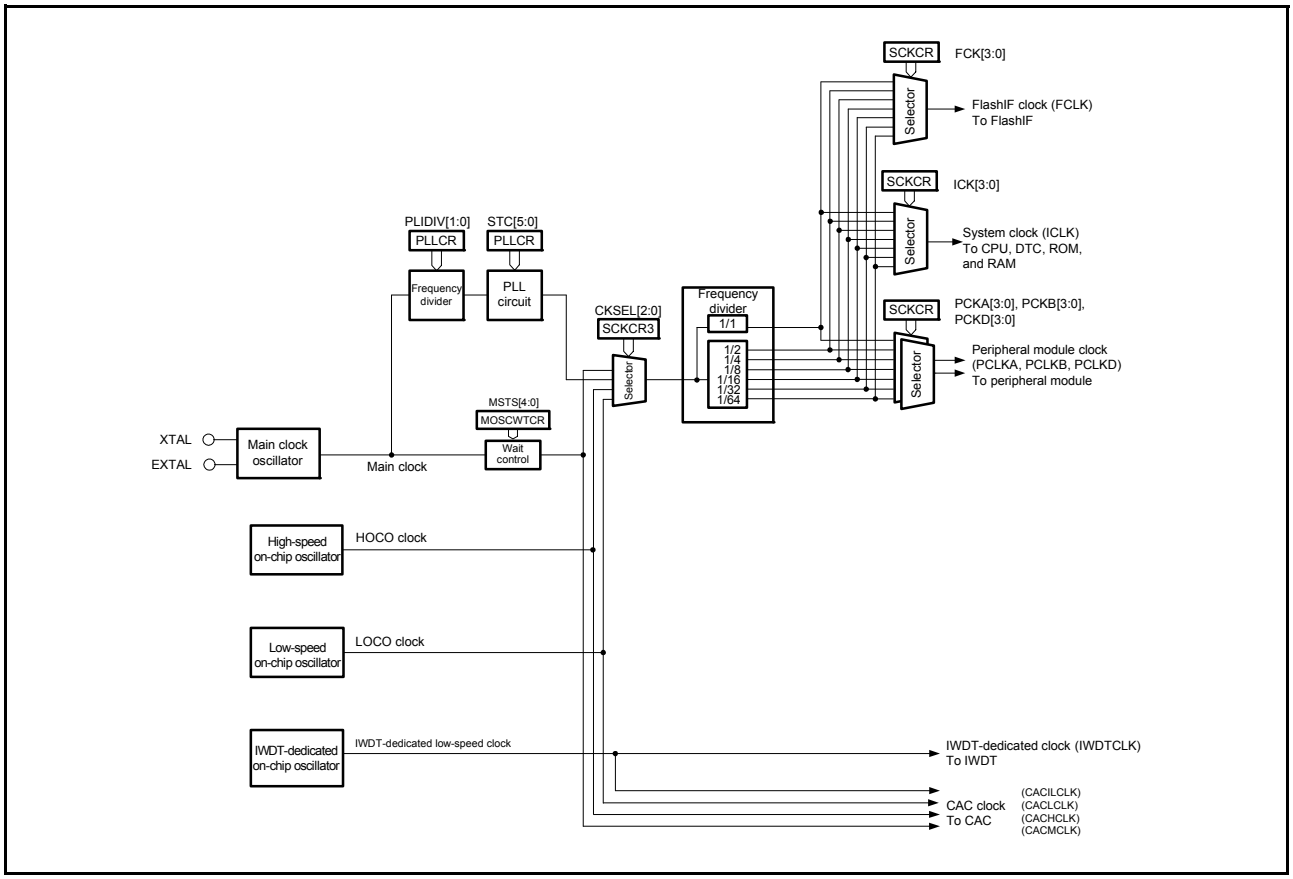
Item	Specification
High-speed on-chip oscillator (HOCO)	Oscillation frequency: 32 MHz

•Page 138, Figure 9.1 Block Diagram of Clock Generation Circuit is corrected as follows:

Before correction



Corrections



•Page 142, “System Clock Control Register 3 (SCKCR3)” is corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10 to b8	CKSEL[2:0] *1	Clock Source Select	b10 b8 0 0 0: LOCO 0 1 0: Main clock oscillator 1 0 0: PLL circuit Settings other than above are prohibited.	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Corrections

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10 to b8	CKSEL[2:0] *1	Clock Source Select	b10 b8 0 0 0: LOCO 0 0 1: HOCO 0 1 0: Main clock oscillator 1 0 0: PLL circuit Settings other than above are prohibited.	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

•Page 147, the following register description is added after the description of the IWDT-Dedicated On-Chip Oscillator Control Register (ILOCOCR):
9.2.8 High-Speed On-Chip Oscillator Control Register (HOCOCR)
9.2.9 High-Speed On-Chip Oscillator Wait Control Register (HOCOWTCR)

The number of the following sections are changed according to the additional description above.

Before correction

9.2.8 Oscillation Stabilization Flag Register (OSCOVFSR)
9.2.9 Oscillation Stop Detection Control Register (OSTDCR)
9.2.10 Oscillation Stop Detection Status Register (OSTDSR)
9.2.11 Main Clock Oscillator Wait Control Register (MOSCWTCR)
9.2.12 Main Clock Oscillator Forced Oscillation Control Register (MOFCR)
9.2.13 Memory Wait Cycle Setting Register (MEMWAIT)

Corrections

9.2.10 Oscillation Stabilization Flag Register (OSCOVFSR)
9.2.11 Oscillation Stop Detection Control Register (OSTDCR)
9.2.12 Oscillation Stop Detection Status Register (OSTDSR)
9.2.13 Main Clock Oscillator Wait Control Register (MOSCWTCR)
9.2.14 Main Clock Oscillator Forced Oscillation Control Register (MOFCR)
9.2.15 Memory Wait Cycle Setting Register (MEMWAIT)

•Page 148, “Oscillation Stabilization Flag Register (OSCOVFSR)” is corrected as follows:
The HCOVF bit is added in b3.

Before correction

Address(es): 0008 003Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PLOVF	—	MOOVF
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MOOVF	Main Clock Oscillation Stabilization Flag	0: Main clock is stopped 1: Oscillation is stable and the clock can be used as the system clock*1	R
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	PLOVF	PLL Clock Oscillation Stabilization Flag	0: PLL is stopped or not stabilized 1: Oscillation is stable and the clock can be used as the system clock	R
b3	—	Reserved	This bit is read as 0 and cannot be modified.	R
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Corrections

Address(es): 0008 003Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	HCOVF	PLOVF	—	MOOVF
Value after reset:	0	0	0	0	0/1*1	0	0	0

Note 1. The HCOVF value after a reset is 1 when the HOCO oscillation enable bit in option function selection register 1 (OFS1.HOCOEN) is 0. The HCOVF value after a reset is 0 when the OFS1.HOCOEN bit is 1.

Bit	Symbol	Bit Name	Description	R/W
b0	MOOVF	Main Clock Oscillation Stabilization Flag	0: Main clock is stopped 1: Oscillation is stable and the clock can be used as the system clock*1	R
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	PLOVF	PLL Clock Oscillation Stabilization Flag	0: PLL is stopped or not stabilized 1: Oscillation is stable and the clock can be used as the system clock	R
b3	HCOVF	HOCO Clock Oscillation Stabilization Flag	0: HOCO is stopped or not stabilized 1: Oscillation is stable and the clock can be used as the system clock*1	R
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

(Additional description)

HCOVF Flag (HOCO Clock Oscillation Stabilization Flag)

This flag indicates whether oscillation of the HOCO clock is stable.

[Setting condition]

- After the HOCOCR.HCSTP bit is set to 0 (HOCO is operating) when the HCSTP bit is 1 (HOCO is stopped), supply of the HOCO clock is started to the MCU internally.

[Clearing condition]

- After the HOCOCR.HCSTP bit is set to 1, the processing to stop the oscillation of the HOCO is completed.

•Page 158, the following description is added to “Oscillation Stop Detection Interrupts”.

When the PLL detects an oscillation stop and is running at its own oscillation frequency, this indicates the occurrence of some system failure. An emergency measure should be taken to handle the failure.

•Page 159, “Internal Clock” is corrected as follows:

Before correction

Clock sources of internal clock signals are the main clock, LOCO clock, PLL clock, and dedicated low-speed clock for the IWDT. The internal clocks listed below are produced from these sources.

Corrections

Clock sources of internal clock signals are the main clock, HOCO clock, LOCO clock, PLL clock, and dedicated low-speed clock for the IWDT. The internal clocks listed below are produced from these sources.

•Page 159, “CAC Clock” is corrected as follows:

Before correction

The CACCLK clocks include CACMCLK which is generated by the main clock oscillator, CACLCLK which is generated by the low-speed on-chip oscillator, and CACILCLK which is generated by the IWDT-dedicated on-chip oscillator.

Corrections

The CACCLK clocks include CACMCLK which is generated by the main clock oscillator, CACHCLK which is generated by the high-speed on-chip oscillator, CACLCLK which is generated by the low-speed on-chip oscillator, and CACILCLK which is generated by the IWDT-dedicated on-chip oscillator.

•Page 160, “Notes on Resonator Connection Pins” is added as follows:

9.7.4 Notes on Resonator Connection Pins

When the main clock is not used, the EXTAL and XTAL pins can be used as general ports P36 and P37. When using these pins as general ports, be sure to stop the main clock (MOSCCR.MOSTP = 1). However, do not use the EXTAL and XTAL pins as general ports P36 and P37 in a system that uses the main clock.
When the main clock is used, do not set P36 and P37 to output.

•Page 161, Table 10.1 CAC Specifications is corrected as follows:

Before correction

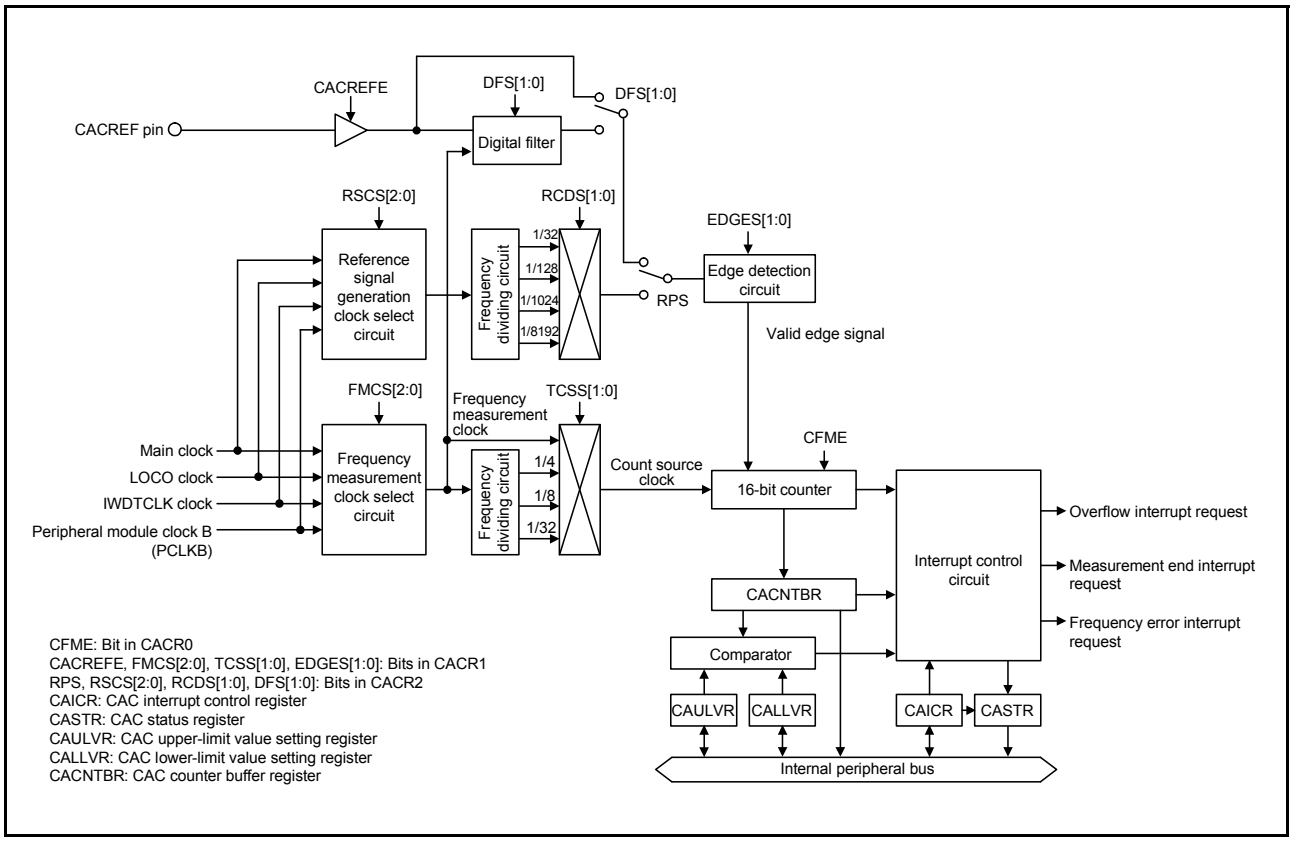
Item	Description
Measurement target clocks	The frequency of the following clocks can be measured. • Main clock • LOCO clock • IWDTCLK clock • Peripheral module clock B (PCLKB)
Measurement reference clocks	• External clock input to the CACREF pin • Main clock • LOCO clock • IWDTCLK clock • Peripheral module clock B (PCLKB)

Corrections

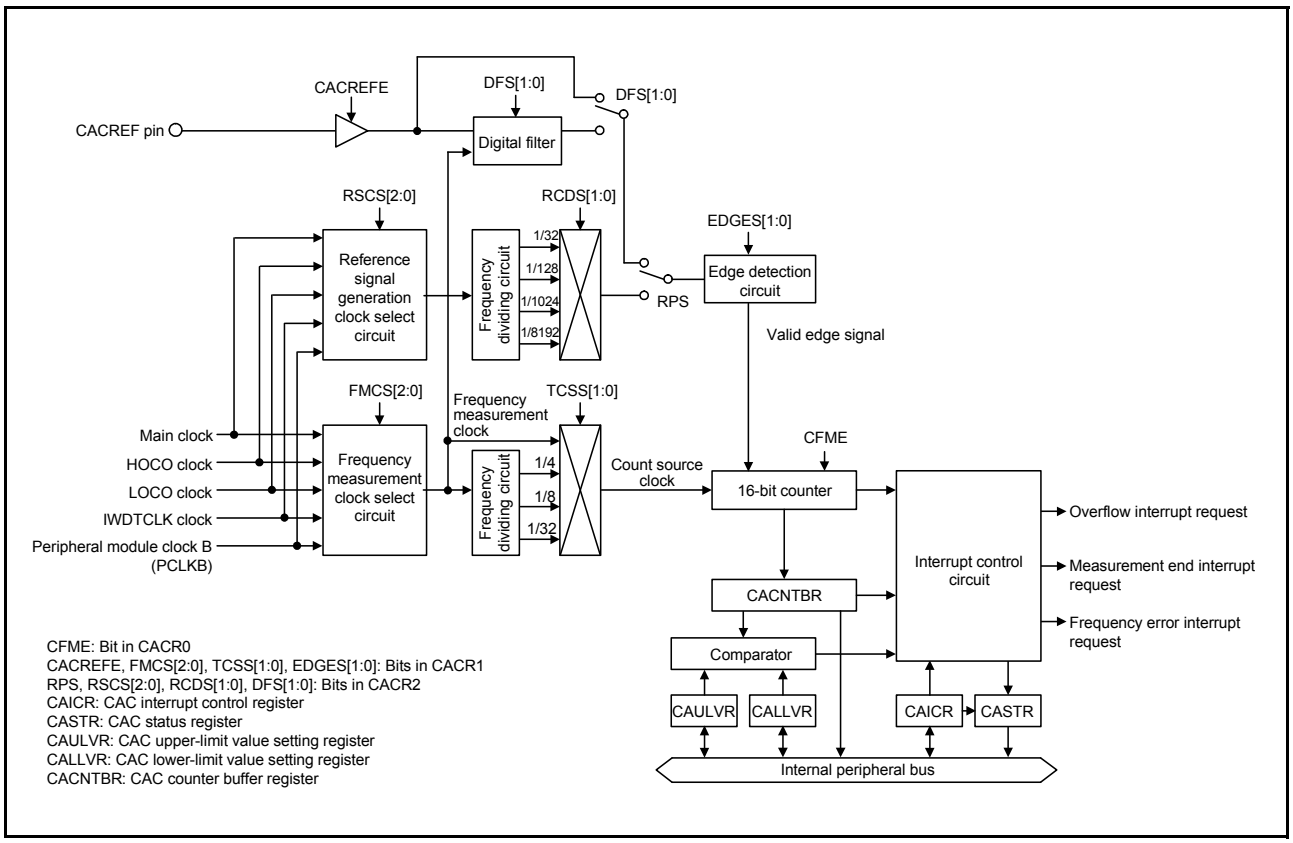
Item	Description
Measurement target clocks	The frequency of the following clocks can be measured. • Main clock • HOCO clock • LOCO clock • IWDTCLK clock • Peripheral module clock B (PCLKB)
Measurement reference clocks	• External clock input to the CACREF pin • Main clock • HOCO clock • LOCO clock • IWDTCLK clock • Peripheral module clock B (PCLKB)

•Page 161, Figure 10.1 CAC Block Diagram is corrected as follows:

Before correction



Corrections



•Page 163, “CAC Control Register 1 (CACR1)” is corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description	R/W
b3 to b1	FMCS[2:0]	Measurement Target Clock Select	b3 b1 0 0 0: Main clock 0 1 1: LOCO clock 1 0 0: IWDTCLK clock 1 0 1: Peripheral module clock B (PCLKB) Settings other than above are prohibited.	R/W

Corrections

Bit	Symbol	Bit Name	Description	R/W
b3 to b1	FMCS[2:0]	Measurement Target Clock Select	b3 b1 0 0 0: Main clock 0 1 0: HOCO clock 0 1 1: LOCO clock 1 0 0: IWDTCLK clock 1 0 1: Peripheral module clock B (PCLKB) Settings other than above are prohibited.	R/W

•Page 172, Table 11.2 Operating Conditions of Each Power Consumption Mode is corrected as follows:

Before correction

	Sleep Mode	Deep Sleep Mode	Software Standby Mode
Entry trigger	Control register + instruction	Control register + instruction	Control register + instruction
Exit trigger	Interrupt	Interrupt	Interrupt*1
After exiting from each mode, CPU begins from*2	Interrupt handling	Interrupt handling	Interrupt handling
Main clock oscillator	Operating possible	Operating possible	Stopped
Low-speed on-chip oscillator	Operating possible	Operating possible	Stopped
IWDT-dedicated on-chip oscillator	Operating possible*3	Operating possible*3	Operating possible*3
PLL	Operating possible	Operating possible	Stopped
CPU	Stopped (Retained)	Stopped (Retained)	Stopped (Retained)
RAM0 (0000 0000h to 0000 27FFh)	Operating possible (Retained)	Stopped (Retained)	Stopped (Retained)
DTC	Operating possible*5	Stopped (Retained)	Stopped (Retained)
Flash memory	Operating	Stopped (Retained)	Stopped (Retained)
Independent watchdog timer (IWDT)	Operating possible*3	Operating possible*3	Operating possible*3
Voltage detection circuit (LVD)	Operating possible	Operating possible	Operating possible
Power-on reset circuit	Operating	Operating	Operating
Peripheral modules	Operating possible	Operating possible	Stopped (Retained)*4
I/O ports	Operating	Operating	Retained
Comparator C	Operating possible	Operating possible	Operating possible*6

Corrections

	Sleep Mode	Deep Sleep Mode	Software Standby Mode
Entry trigger	Control register + instruction	Control register + instruction	Control register + instruction
Exit trigger	Interrupt	Interrupt	Interrupt*1
After exiting from each mode, CPU begins from*2	Interrupt handling	Interrupt handling	Interrupt handling
Main clock oscillator	Operating possible	Operating possible	Stopped
High-speed on-chip oscillator	Operating possible	Operating possible	Stopped
Low-speed on-chip oscillator	Operating possible	Operating possible	Stopped
IWDT-dedicated on-chip oscillator	Operating possible*3	Operating possible*3	Operating possible*3
PLL	Operating possible	Operating possible	Stopped
CPU	Stopped (Retained)	Stopped (Retained)	Stopped (Retained)
RAM0 (0000 0000h to 0000 27FFh, 0000 4000h to 0000 4A7Fh)	Operating possible (Retained)	Stopped (Retained)	Stopped (Retained)
DTC	Operating possible*5	Stopped (Retained)	Stopped (Retained)
Flash memory	Operating	Stopped (Retained)	Stopped (Retained)
Independent watchdog timer (IWDT)	Operating possible*3	Operating possible*3	Operating possible*3
Voltage detection circuit (LVD)	Operating possible	Operating possible	Operating possible
Power-on reset circuit	Operating	Operating	Operating
Peripheral modules	Operating possible	Operating possible	Stopped (Retained)*4
I/O ports	Operating	Operating	Retained
Comparator C	Operating possible	Operating possible	Operating possible*6

•Page 178, “Module Stop Control Register C (MSTPCRC)” is corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPC0	RAM0 Module Stop*1	Target module: RAM0 (0000 0000h to 0000 27FFh) 0: RAM0 operating 1: RAM0 stopped	R/W

Corrections

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPC0	RAM0 Module Stop*1	Target module: RAM0 (0000 0000h to 0000 27FFh, 0000 4000h to 0000 4A7Fh) 0: RAM0 operating 1: RAM0 stopped	R/W

•Page 192, Table 12.1 Association between PRCR Bits and Registers to be Protected is corrected as follows:

Before correction

PRCR Bit	Register to be Protected
PRC0	· Registers related to the clock generation circuit: SCKCR, SCKCR3, PLLCR, PLLCR2, MOSCCR, LOCOCR, ILOCOCR, OSTDCR, OSTDSR, MEMWAITLOCOTRR, ILOCOTRR, HOCOTRR0, HOCOTRR3
PRC1	· Register related to the operating modes: SYSCR1 · Registers related to low power consumption functions: SBYCR, MSTPCRA, MSTPCRB, MSTPCRC, OPCCR · Registers related to the clock generation circuit: MOFCR, MOSCWTCR · Software reset register: SWRR
PRC3	· Registers related to the LVD: LVCMPCR, LVDLVLR, LVD1CR0, LVD1CR1, LVD1SR, LVD2CR0, LVD2CR1, LVD2SR

Corrections

PRCR Bit	Register to be Protected
PRC0	· Registers related to the clock generation circuit: SCKCR, SCKCR3, PLLCR, PLLCR2, MOSCCR, LOCOCR, ILOCOCR, HOCOCR, OSTDCR, OSTDSR, MEMWAIT
PRC1	· Register related to the operating modes: SYSCR1 · Registers related to low power consumption functions: SBYCR, MSTPCRA, MSTPCRB, MSTPCRC, OPCCR · Registers related to the clock generation circuit: MOFCR, MOSCWTCR · Software reset register: SWRR
PRC2	· Register related to the clock generation circuit: HOCOWTCR
PRC3	· Registers related to the LVD: LVCMPCR, LVDLVLR, LVD1CR0, LVD1CR1, LVD1SR, LVD2CR0, LVD2CR1, LVD2SR

•Page 193, “Protect Register (PRCR)” is corrected as follows:

Before correction

Address(es): 0008 03FEh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	PRC3	—	PRC1	PRC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	PRC0	Protect Bit 0	Enables writing to the registers related to the clock generation circuit and the flash memory. 0: Write disabled 1: Write enabled	R/W
b1	PRC1	Protect Bit 1	Enables writing to the registers related to operating modes, low power consumption functions, the clock generation circuit, and software reset. 0: Write disabled 1: Write enabled	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	PRC3	Protect Bit 3	Enables writing to the registers related to the LVD. 0: Write disabled 1: Write enabled	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	PRC Key Code	These bits control permission and prohibition of writing to the PRCR register. To modify the PRCR register, write A5h to the 8 higher-order bits and the desired value to the 8 lower-order bits as a 16-bit unit.	R/W*1

Note 1. Write data is not retained.

PRCi Bits (Protect Bit i) (i = 0, 1, 3)

These bits enable or disable writing to the corresponding registers to be protected.

Setting the PRCi bits to 1 and 0 enable and disable writing to the corresponding registers to be protected, respectively.

Corrections

Address(es): 0008 03FEh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	PRC3	PRC2	PRC1	PRC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	PRC0	Protect Bit 0	Enables writing to the registers related to the clock generation circuit. 0: Write disabled 1: Write enabled	R/W
b1	PRC1	Protect Bit 1	Enables writing to the registers related to operating modes, low power consumption functions, the clock generation circuit, and software reset. 0: Write disabled 1: Write enabled	R/W
b2	PRC2	Protect Bit 2	Enables writing to the registers related to the clock generation circuit. 0: Write disabled 1: Write enabled	R/W
b3	PRC3	Protect Bit 3	Enables writing to the registers related to the LVD. 0: Write disabled 1: Write enabled	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	PRC Key Code	These bits control permission and prohibition of writing to the PRCR register. To modify the PRCR register, write A5h to the 8 higher-order bits and the desired value to the 8 lower-order bits as a 16-bit unit.	R/W*1

Note 1. Write data is not retained.

PRCi Bits (Protect Bit i) (i = 0 to 3)

These bits enable or disable writing to the corresponding registers to be protected.

Setting the PRCi bits to 1 and 0 enable and disable writing to the corresponding registers to be protected, respectively.

•Page 239, Table 15.2 Addresses Assigned for Each Bus is corrected as follows:

Before correction

Address	Bus	Area
0000 0000h to 0000 27FFh	Memory bus 1	RAM
0000 2800h to 0007 FFFFh		Reserved area
0008 0000h to 0008 7FFFh	Internal peripheral bus 1	Peripheral I/O registers
0008 8000h to 0009 FFFFh	Internal peripheral bus 2	
000A 0000h to 000B FFFFh	Internal peripheral bus 3	
000C 0000h to 000D FFFFh	Internal peripheral bus 4	
0010 0000h to 00FF FFFFh	Internal peripheral bus 6	Flash control module
8000 0000h to FEFF FFFFh	Memory bus 2	ROM (for reading only)
FF00 0000h to FFFF FFFFh		

Corrections

Address	Bus	Area
0000 0000h to 0000 27FFh	Memory bus 1	RAM
0000 2800h to 0000 3FFFh		Reserved area
0000 4000h to 0000 4A7Fh		RAM
0000 4A80h to 0007 FFFFh		Reserved area
0008 0000h to 0008 7FFFh	Internal peripheral bus 1	Peripheral I/O registers
0008 8000h to 0009 FFFFh	Internal peripheral bus 2	
000A 0000h to 000B FFFFh	Internal peripheral bus 3	
000C 0000h to 000D FFFFh	Internal peripheral bus 4	
0010 0000h to 00FF FFFFh	Internal peripheral bus 6	Flash control module
8000 0000h to FEFF FFFFh	Memory bus 2	ROM (for reading only)
FF00 0000h to FFFF FFFFh		

•Page 303, Table 18.1 Specifications of I/O Ports is corrected as follows:

Before correction

Port	Package		Package		Package	
	64 Pins	Number of Pin	52 Pins	Number of Pin	48 Pins	Number of Pin
PORT0	P00 to P02	3	P02	1	None	None
PORT1	P10, P11	2	P10, P11	2	P10, P11	2
PORT2	P22 to P24	3	P22 to P24	3	P22 to P24	3
PORT3	P30 to P33	4	P33	1	None	None
PORT4	P40 to P47	8	P40 to P47	8	P40 to P47	8
PORT7	P70 to P76	7	P70 to P76	7	P70 to P76	7
PORT9	P91 to P94	4	P93, P94	2	P93, P94	2
PORTA	PA2 to PA5	4	PA2, PA3, PA5	3	PA2, PA3	2
PORTB	PB0 to PB7	8	PB0 to PB7	8	PB0 to PB6	7
PORTD	PD3 to PD7	5	PD3 to PD6	4	PD3 to PD6	4
PORTE	PE2	1	PE2	1	PE2	1
	Total of Pins	49	Total of Pins	39	Total of Pins	36

Corrections

Port	Package		Package		Package	
	64 Pins	Number of Pin	52 Pins	Number of Pin	48 Pins	Number of Pin
PORT0	P00 to P02	3	P02	1	None	None
PORT1	P10, P11	2	P10, P11	2	P10, P11	2
PORT2	P22 to P24	3	P22 to P24	3	P22 to P24	3
PORT3	P30 to P33, P36, P37	6	P33, P36, P37	3	P36, P37	2
PORT4	P40 to P47	8	P40 to P47	8	P40 to P47	8
PORT7	P70 to P76	7	P70 to P76	7	P70 to P76	7
PORT9	P91 to P94	4	P93, P94	2	P93, P94	2
PORTA	PA2 to PA5	4	PA2, PA3, PA5	3	PA2, PA3	2
PORTB	PB0 to PB7	8	PB0 to PB7	8	PB0 to PB6	7
PORTD	PD3 to PD7	5	PD3 to PD6	4	PD3 to PD6	4
PORTE	PE2	1	PE2	1	PE2	1
	Total of Pins	51	Total of Pins	42	Total of Pins	38

•Page 304, Table 18.2 Port Functions is corrected as follows:

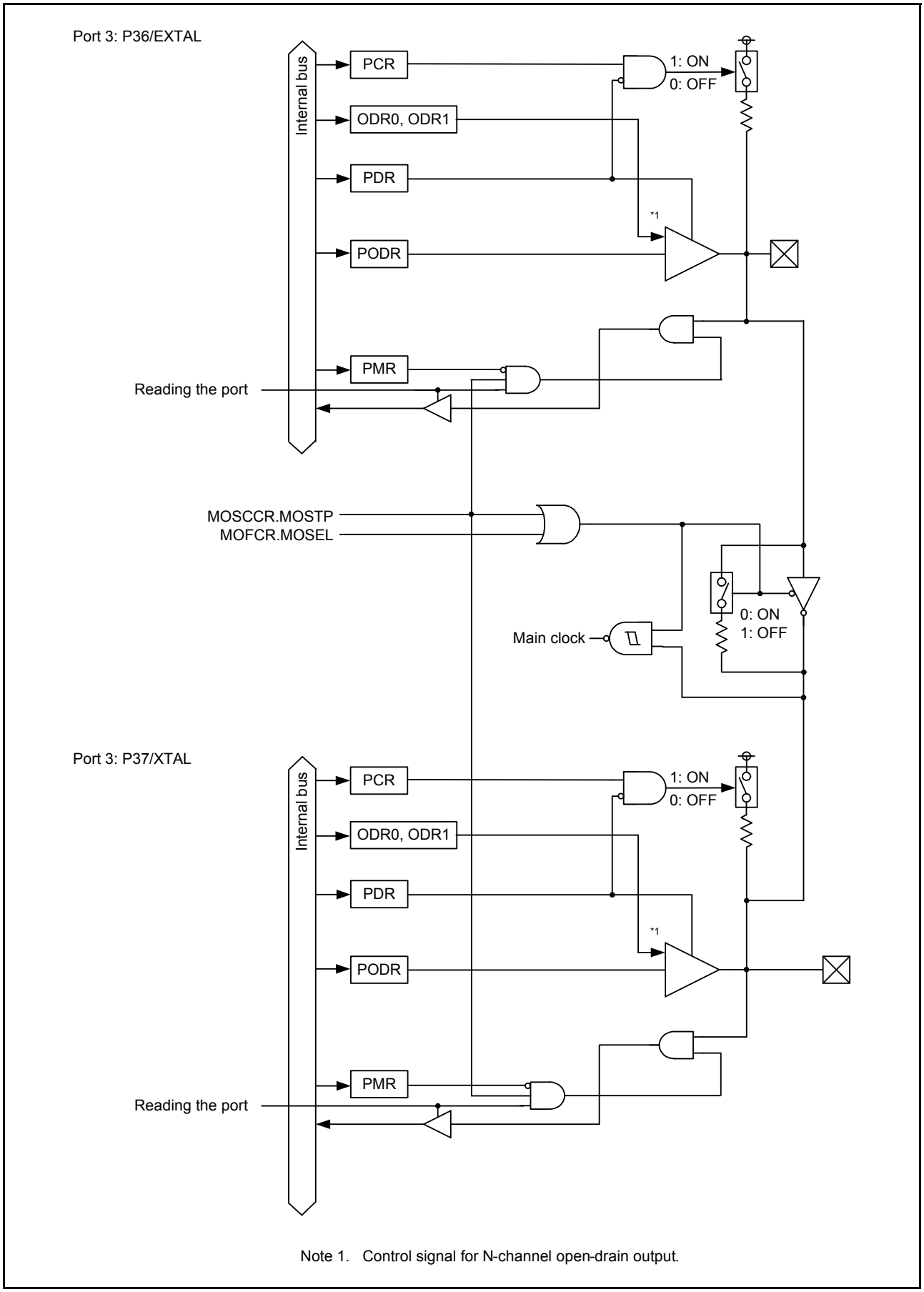
Before correction

Port	Pin	Input Pull-up	Open Drain Output	Drive Capacity Switching	High current pin	5-V Tolerant
PORT3	P30 to P33	○	○	○	—	—

Corrections

Port	Pin	Input Pull-up	Open Drain Output	Drive Capacity Switching	High current pin	5-V Tolerant
PORT3	P30 to P33, P36, P37	○	○	○	—	—

•Page 306, Figure 18.3 I/O Port Configuration (3) is added as follows:



Note 1. Control signal for N-channel open-drain output.

Figure 18.3 I/O Port Configuration (3)

•Page 312, “Open Drain Control Register 1 (ODR1)” is corrected as follows:
Address “PORT3.ODR1 0008 C087h” is added.

Before correction

Address(es): PORT2.ODR1 0008 C085h, PORT7.ODR1 0008 C08Fh, PORT9.ODR1 0008 C093h, PORTA.ODR1 0008 C095h, PORTB.ODR1 0008 C097h, PORTD.ODR1 0008 C09Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset:	0	0	0	0	0	0	0	0

Corrections

Address(es): PORT2.ODR1 0008 C085h, PORT3.ODR1 0008 C087h, PORT7.ODR1 0008 C08Fh, PORT9.ODR1 0008 C093h, PORTA.ODR1 0008 C095h, PORTB.ODR1 0008 C097h, PORTD.ODR1 0008 C09Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset:	0	0	0	0	0	0	0	0

•Page 317, Table 18.6 Unused Pin Configuration is corrected as follows:

Before correction

Pin Name	Description
MD	(Always used as mode pins)
RES#	Connect this pin to VCC via a pull-up resistor.
PE2/NMI	Connect this pin to VCC via a pull-up resistor.
EXTAL	EXTAL (Always used as clock pin)
XTAL	Leave XTAL pin open.
Ports 0 to 4, 7, 9 Ports A, B, D	• If the direction setting is for input (PORTn.PDR = 0), the corresponding pin is connected to VCC (pulled up) via a resistor or to VSS (pulled down) via a resistor.*1 • If the direction setting is for output (PORTn.PDR = 1), the pin is released.*1, *2

Corrections

Pin Name	Description
MD	(Always used as mode pins)
RES#	Connect this pin to VCC via a pull-up resistor.
PE2/NMI	Connect this pin to VCC via a pull-up resistor.
P36/EXTAL	When the main clock is not used, set the MOSCCR.MOSTP bit to 1 (general port P36). When this pin is not used as port P36 either, it is configured in the same way as port 0 to 4, 7, 9, A, B, D.
P37/XTAL	When the main clock is not used, set the MOSCCR.MOSTP bit to 1 (general port P37). When this pin is not used as port P37 either, it is configured in the same way as port 0 to 4, 7, 9, A, B, D. When the external clock is input to the EXTAL pin, leave this pin open.
Ports 0 to 4, 7, 9 Ports A, B, D	• If the direction setting is for input (PORTn.PDR = 0), the corresponding pin is connected to VCC (pulled up) via a resistor or to VSS (pulled down) via a resistor.*1 • If the direction setting is for output (PORTn.PDR = 1), the pin is released.*1, *2

•Page 1102, Table 1.1 Unused Pin Configuration is corrected as follows:

Before correction

Port Name (Pin Name)	Operating Mode According to Registers Setting	Reset	Software Standby Mode
P30 to P33	All	Hi-Z	Keep-O

Corrections

Port Name (Pin Name)	Operating Mode According to Registers Setting	Reset	Software Standby Mode
P30 to P33, P36, P37	All	Hi-Z	Keep-O

<Corrections on Erroneous Descriptions>

•Page 34, the description in “Features” is corrected as follows:

Before correction

■ **Up to 5 communications channels**

Corrections

■ **Up to 4 communications channels**

Before correction

■ **Up to 16 extended-function timers**

Corrections

■ **Up to 12 extended-function timers**

Before correction

PLQP0064KB-A 10 × 10 mm, 0.5 mm pitch
PLQP0052JA-A 10 × 10 mm, 0.65 mm pitch
PLQP0048KB-A 7 × 7 mm, 0.5 mm pitch

Corrections

PLQP0064KB-C 10 × 10 mm, 0.5 mm pitch
PLQP0052JA-B 10 × 10 mm, 0.65 mm pitch
PLQP0048KB-B 7 × 7 mm, 0.5 mm pitch

•Page 137, Table 9.1 Specifications of Clock Generation is corrected as follows:

Before correction

Item	Specification
Uses	• Generates the system clock (ICLK) to be supplied to the CPU, DTC, ROM, and RAM. • Generates the peripheral module clocks (PCLKA, PCLKB, and PCLKD) to be supplied to peripheral modules. The peripheral module clock used as the operating clock is PCLKD for S12AD, and PCLKA and PCLKB are for other modules. • Generates the FlashIF clock (FCLK) to be supplied to the FlashIF. • Generates the CAC clock (CACCLK) to be supplied to the CAC. • Generates the IWDT-dedicated low-speed clock (IWDTCLK) to be supplied to the IWDT.

Corrections

Item	Specification
Uses	• Generates the system clock (ICLK) to be supplied to the CPU, DTC, ROM, and RAM. • Generates the peripheral module clocks (PCLKA, PCLKB, and PCLKD) to be supplied to peripheral modules. The peripheral module clock PCLKA is the operating clock for the MTU3, the peripheral module clock PCLKD is for the S12AD, and PCLKB is for other modules. • Generates the FlashIF clock (FCLK) to be supplied to the FlashIF. • Generates the CAC clock (CACCLK) to be supplied to the CAC. • Generates the IWDT-dedicated low-speed clock (IWDTCLK) to be supplied to the IWDT.

•Page 186, Note 1 is added to the description of “Entry to Deep Sleep Mode”.

Before correction

When a WAIT instruction is executed with the MSTPCRC.DSLPE bit set to 1, the MSTPCRA.MSTPA28 bit set to 1, and the SBYCR.SSBY bit cleared to 0, a transition to deep sleep mode is made.

Corrections

When a WAIT instruction is executed with the MSTPCRC.DSLPE bit set to 1, the MSTPCRA.MSTPA28 bit set to 1, and the SBYCR.SSBY bit cleared to 0, a transition to deep sleep mode is made.*1

Note 1. Transition to deep sleep mode might not be possible, depending on the operating state of the DTC. Before setting the MSTPCRA.MSTPA28 bit to 1, set the DTCST.DTCST bit of the DTC to 0 to avoid activating the DTC.

•Page 234, Note 1 is added to the description of “External Pin Interrupts”.

Before correction

2. Clear the IRQFLTE0.FLTENi bit (i = 0 to 5) to 0 (digital filter disabled).
3. Set the digital filter sampling clock with the IRQFLTC0.FCLKSELi[1:0] bits.
   :
7. Set the IRQFLTE0.FLTENi bit to 1 (digital filter enabled).

Corrections

2. Clear the IRQFLTE0.FLTENi bit (i = 0 to 5) to 0 (digital filter disabled).*1
3. Set the digital filter sampling clock with the IRQFLTC0.FCLKSELi[1:0] bits.*1
   :
7. Set the IRQFLTE0.FLTENi bit to 1 (digital filter enabled).*1

Note 1. To use the digital filter function, settings must be made beforehand.

•Page 235, Note 1 is added to the description of “Non-maskable Interrupt Operation”.

Before correction

2. To use the NMI pin, clear the NMIFLTE.NFLTEN bit to 0 (digital filter disabled).
3. To use the NMI pin, set the digital filter sampling clock with the NMIFLTC.NFCLKSEL[1:0] bits.
   :
6. To use the NMI pin, set the NMIFLTE.NFLTEN bit to 1 (digital filter enabled).

Corrections

2. To use the NMI pin, clear the NMIFLTE.NFLTEN bit to 0 (digital filter disabled).*1
3. To use the NMI pin, set the digital filter sampling clock with the NMIFLTC.NFCLKSEL[1:0] bits.*1
   :
6. To use the NMI pin, set the NMIFLTE.NFLTEN bit to 1 (digital filter enabled).*1

Note 1. To use the digital filter function, settings must be made beforehand.

•Page 301, “Low Power Consumption Function” is corrected as follows:

Before correction

Before making a transition to the module stop state or software standby mode, set the DTCST.DTCST bit to 0 (DTC module stop), and then perform the following.

Corrections

Before making a transition to the module stop state, deep sleep mode, or software standby mode, set the DTCST.DTCST bit to 0 (DTC module stop), and then perform the following.

(2) Deep Sleep Mode

Make settings according to the procedure under section 11.6.2.1, Entry to Deep Sleep Mode, in section 11, Low Power Consumption.

If DTC transfer operations are in progress at the time the WAIT instruction is executed, the transition to deep sleep mode follows the completion of DTC transfer.

The DTC is released from the module stop state by writing 0 to the MSTPCRA.MSTPA28 bit following recovery from deep sleep mode.

•Page 303, The description of “Overview” in section 18 I/O Ports is corrected as follows:

Before correction

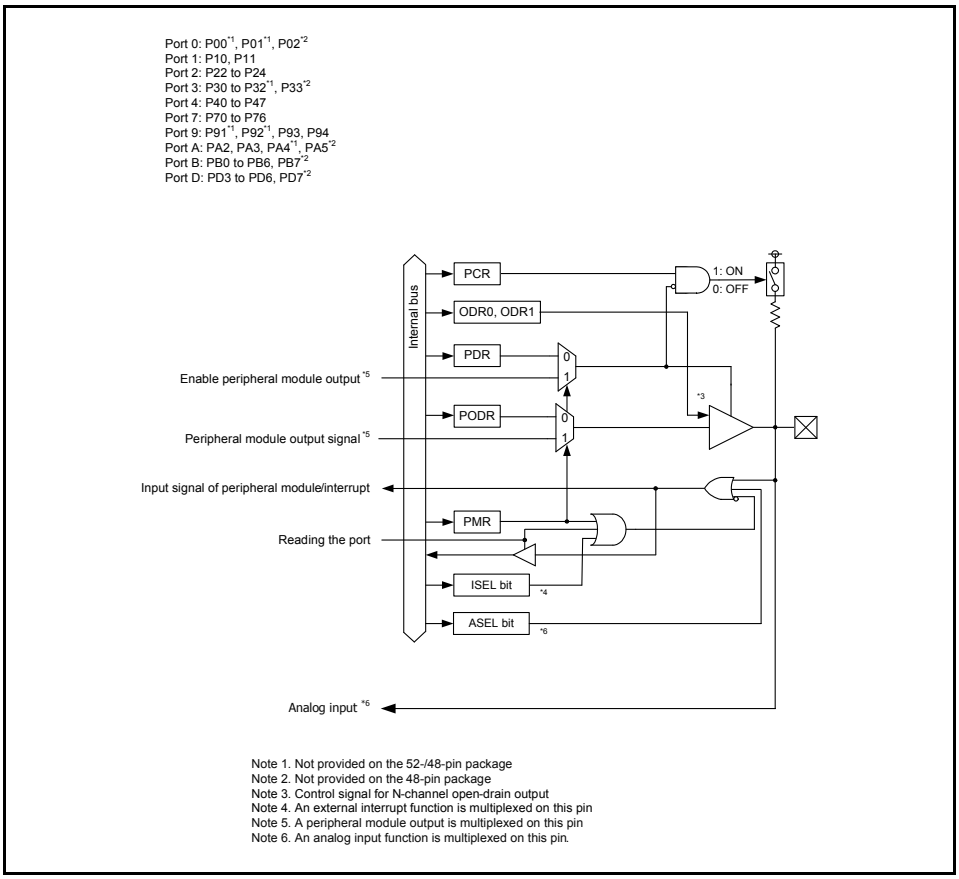
The I/O ports function as a general I/O port, an I/O pin of a peripheral module, an input pin for an interrupt, or a bus control pin.

Corrections

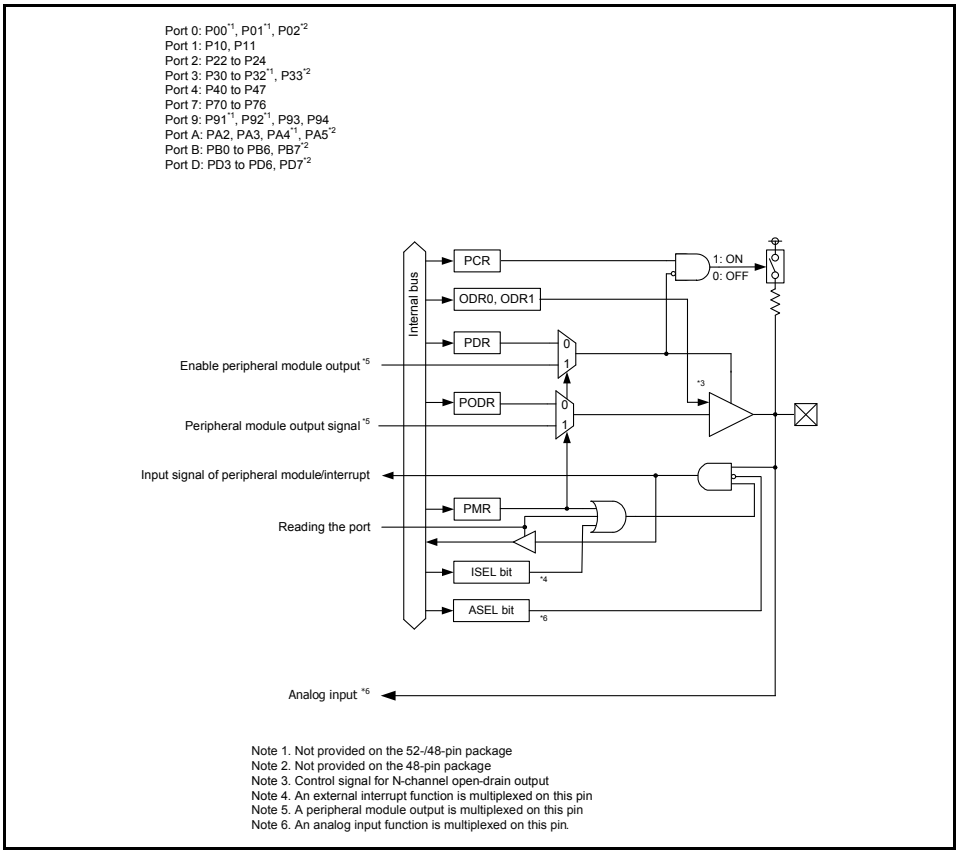
The I/O ports function as a general I/O port, an I/O pin of a peripheral module, or an input pin for an interrupt

•Page 305, Figure 18.1 I/O Port Configuration (1) is corrected as follows:

Before correction

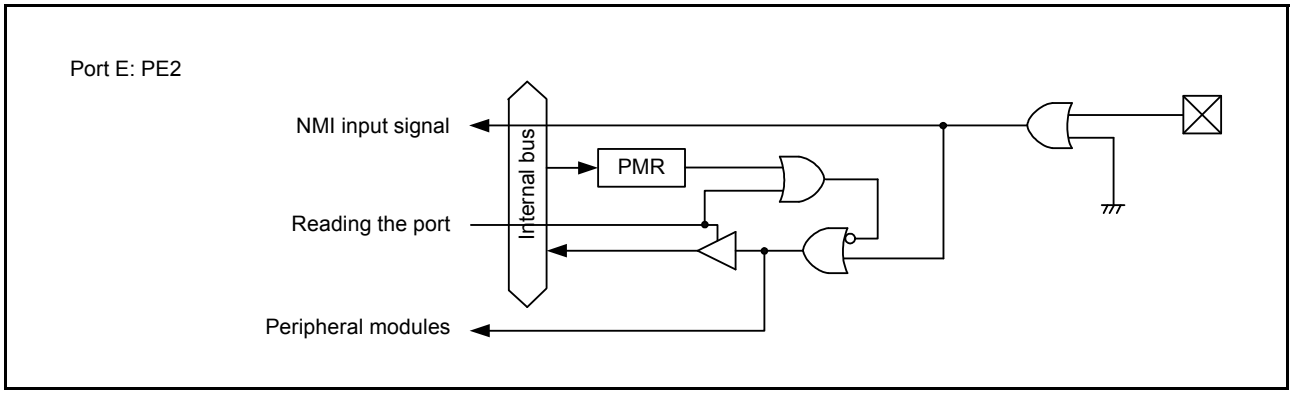


Corrections

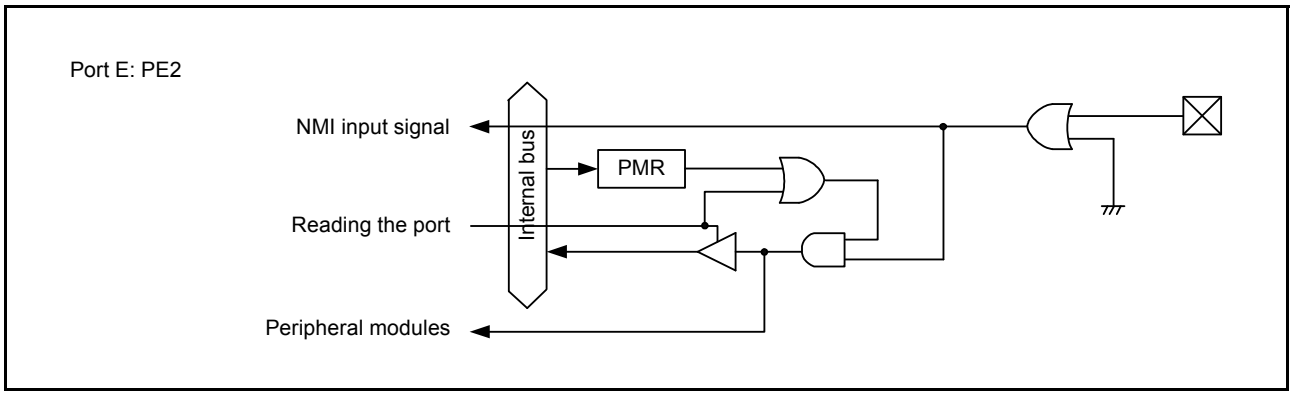


•Page 306, Figure 18.2 I/O Port Configuration (2) is corrected as follows:

Before correction



Corrections



•Page 307, “Port Direction Register (PDR)” is corrected as follows:

Before correction

Write 1 (output) to each bit of PDR corresponding to port m that does not exist.

Corrections

Each bit of PDR corresponding to port m that does not exist is reserved. Make settings according to the description in section 18.4, Initialization of the Port Direction Register (PDR).

The PORTE.PDR.B2 bit is reserved, because the PE2 pin is input only. A reserved bit is read as 0. The write value should be 0.

•Page 336, Table 20.1 MTU Specifications is corrected as follows:

Before correction

Count clock	11 clocks for each channel (14 clocks for MTU0, 12 clocks for MTU1 and MTU2, and 10 clocks for MTU5)

Corrections

Count clock	11 clocks for each channel (14 clocks for MTU0, 12 clocks for MTU2, 10 clocks for MTU5, and four clocks for MTU1 & MTU2 (LWA = 1))

Before correction

[MTU1, MTU2]
- Phase counting mode can be specified independently
- Cascade connection operation available

Corrections

[MTU1, MTU2]
- Phase counting mode can be specified independently
- 32-bit phase counting mode can be specified for interlocked operation of MTU1 and MTU2 (when TMDR3.LWA = 1)
- Cascade connection operation available

•Page 342, the following description is deleted from Table 20.3 Pin Configuration of the MTU.

Channel	Pin Name	I/O	Function
MTU	ADSM1	Output	A/D conversion start request frame synchronization signal 1 output pin

•Page 343 and 344, Table 20.4 CCLR[2:0] (MTU0, MTU3, MTU4) and Table 20.5 CCLR[2:0] (MTU1 and MTU2) in 20.2.2 Timer Control Register 2 (TCR2) are moved to section 20.2.1 Timer Control Register (TCR):

•Page 417, “Cascaded Operation” is corrected as follows:

Before correction

In cascaded operation, two 16-bit counters in different channels are used together as a 32-bit counter.
This function operates when the TCR.TPSC[2:0] bits are set so that the MTU1 count clock is selected to count an overflow/underflow of MTU2.TCNT.
Underflow occurs only when the lower 16-bits of TCNT are in phase counting mode.
Table 20.48 shows the register combinations used in cascaded operation.

Corrections

In cascaded operation, two 16-bit counters in different channels are used together as a 32-bit counter.
There are two functions for connecting MTU1 and MTU2 to use as a 32-bit counter: cascade connection to be set when the MTU1.TMDR3.LWA bit is 0, and cascade connection 32-bit phase counting mode to be set when the MTU1.TMDR3.LWA bit is 1. For details on cascade connection 32-bit phase counting mode, refer to section 20.3.6.2, Cascade Connection 32-Bit Phase Counting Mode. This section describes the cascade connection function to be set when the MTU1.TMDR3.LWA bit is 0.
This function operates when the MTU1.TMDR3.LWA bit is set to 0 and the MTU1.TCR.TPSC[2:0] bits are set so that MTU1.TCNT counts at an overflow/underflow of MTU2.TCNT. Underflow occurs only when the MTU2 to which the lower 16 bits allocated is in phase counting mode.
Table 20.48 shows the register combinations used in cascaded operation.

•Page 427, 20.3.6 Phase Counting Mode is corrected as follows:

Before correction

In phase counting mode, the phase difference between two external input clocks is detected and the corresponding TCNT is incremented or decremented.

Two external clock input pins for each phase counting mode are not affected by the settings of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. In phase counting mode, the external clock pins MTCLKA, MTCLKB, MTCLKC, and MTCLKD are used for two-phase encoder pulse input.

Table 20.51 lists the external clock input pins to be connected in each phase counting mode.

Table 20.51 Clock Input Pins in Phase Counting Mode

	External Clock Input Pins	
Channel	**A-Phase**	**B-Phase**
MTU1	MTCLKA	MTCLKB
MTU2	MTCLKC	MTCLKD

Corrections

There are two phase counting modes: 16-bit phase counting mode in which MTU1 and MTU2 operate independently, and cascade connection 32-bit phase counting mode in which MTU1 and MTU2 are cascaded.

In phase counting mode, the phase difference between two external input clocks is detected and the corresponding TCNT is incremented or decremented.

Two external clock input pins for each phase counting mode are not affected by the settings of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. The two external clock input pins used in 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2 can be selected by MTU1.TMDR3.PHCKSEL. In a phase counting mode other than 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2, MTCLKA and MTCLKB are selected for A-phase and B-phase, respectively. In phase counting mode, the external clock pins MTCLKA, MTCLKB, MTCLKC, and MTCLKD are used for two-phase encoder pulse input.

Table 20.51 lists the external clock input pins to be connected ineach phase counting mode.

Table 20.51 Clock Input Pins in Phase Counting Mode

		External Clock Input Pins	
Phase Counting Mode	**TMDR3.PHCKSEL bit**	**A-Phase**	**B-Phase**
MTU1 16-bit phase counting mode	x (Don't care)	MTCLKA	MTCLKB
MTU2 16-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD
Cascade connection 32-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD

•Page427, 20.3.6.1 Phase Counting Mode is corrected as follows:

Before correction

20.3.6.1 Phase Counting Mode

In phase counting mode, the phase difference between two external input clocks is detected and the 16-bit counter TCNT of the corresponding channel is incremented or decremented.

When phase counting mode is specified, an external clock is selected as the counter input clock and TCNT operates as an up-counter/down-counter regardless of the setting of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. However, the functions of TCR.CCLR[1:0],TIOR, TIER, and TGR are enabled, and input capture/compare match and interrupt functions can be used.

These external input pins can be used for two-phase encoder pulse input.

Corrections

20.3.6.1 16-Bit Phase Counting Mode

When the MTU1.TMDR3.LWA is 0, 16-bit phase counting mode can be set individually for MTU1 and MTU2.
In 16-bit phase counting mode, the phase difference between two external input clocks is detected and the 16-bit counter TCNT of the corresponding channel is incremented or decremented.
When 16-bit phase counting mode is specified, an external clock is selected as the counter input clock and TCNT operates as an up-counter/down-counter regardless of the setting of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. However, the functions of TCR.CCLR[1:0],TIOR, TIER, and TGR are enabled, and input capture/compare match and interrupt functions can be used.
These external input pins can be used for two-phase encoder pulse input.

•Page 428, (1) Example of Phase Counting Mode Setting Procedure is corrected as follows:

Before correction

(1) Example of Phase Counting Mode Setting Procedure

Figure 20.29 shows an example of the phase counting mode setting procedure.

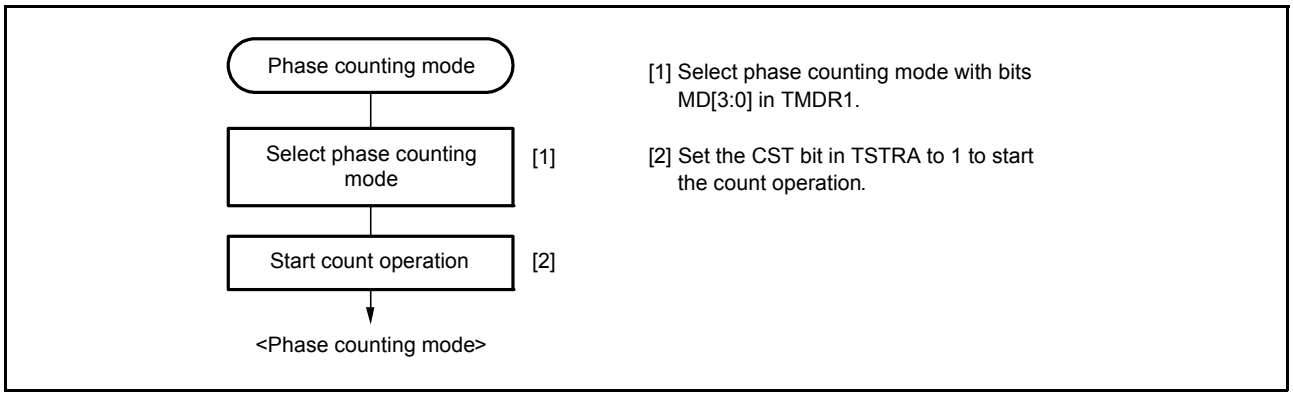


Figure 20.29 Example of Phase Counting Mode Setting Procedure

Corrections

(1) Example of 16-Bit Phase Counting Mode Setting Procedure

Figure 20.29 shows an example of the phase counting mode setting procedure.

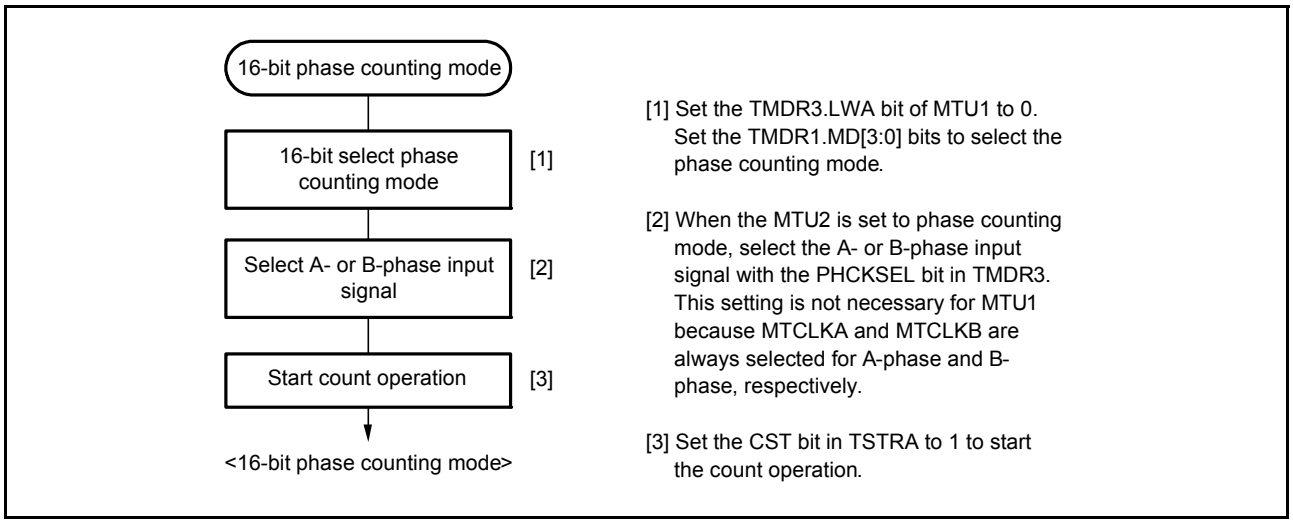


Figure 20.29 Example of 16-Bit Phase Counting Mode Setting Procedure

•Page 429, (2) Example of Phase Counting Mode Operation is corrected as follows:

Before correction

(2) Examples of Phase Counting Mode Operation

In phase counting mode, TCNT is incremented or decremented according to the phase difference between two external clocks. There are five modes according to the count conditions.

Corrections

(2) Examples of 16-Bit Phase Counting Mode Operation

In phase counting mode, TCNT is incremented or decremented according to the phase difference between two external clocks. There are five modes according to the count conditions. Each mode operates under the condition PHCKSEL = 1, which means the phase clock for MTU1 is input from MTCLKA or MTCLKB and that for MTU2 is input from MTCLKC or MTCLKD.

•Page 438, (3) Phase Counting Mode Application Example is corrected as follows:

Before correction

(3) Phase Counting Mode Application Example

Corrections

(3) 16-Bit Phase Counting Mode Application Example

•Page 438, “Cascade Connection 32-Bit Phase Counting Mode” is added.

•Page 479, (a) Register and Counter Miswrite Prevention Function is corrected as follows:

Before correction

22 registers in total
MTU3.TCR, MTU4.TCR, MTU3.TMDR1, MTU4.TMDR1, MTU3.TIORH,
MTU4.TIORH, MTU3.TIORL, MTU4.TIORL, MTU3.TIER, MTU4.TIER,
MTU3.TCNT, MTU4.TCNT, MTU3.TGRA, MTU4.TGRA, MTU3.TGRB, MTU4.TGRB,
TOERA, TOCR1A, TOCR2A, TGCRA, TCDRA, TDDRA

Corrections

24 registers in total
MTU3.TCR, MTU4.TCR, MTU3.TCR2, MTU4.TCR2, MTU3.TMDR1, MTU4.TMDR1, MTU3.TIORH, MTU4.TIORH, MTU3.TIORL, MTU4.TIORL, MTU3.TIER, MTU4.TIER, MTU3.TCNT, MTU4.TCNT, MTU3.TGRA, MTU4.TGRA, MTU3.TGRB, MTU4.TGRB, MTU.TOERA, MTU.TOCR1A, MTU.TOCR2A, MTU.TGCRA, MTU.TCDRA, and MTU.TDDRA

•Page 484, “Synchronous Operation of MTU0 to MTU4” is added.

•Page 489, A/D Conversion Start Request Frame Synchronization Signal is corrected as follows:

Before correction

This function can be used to monitor the generation timing of the A/D conversion start request signal using an external pin. When the A/D conversion request signal to be monitored is selected by the TADSTRGRn register and the ADSMn pin output is enabled by TADSTRGRn.TADSMENn, a pulse signal is output is at the high level when the A/D conversion start request signal is generated, and at the low level in the timer cycle used to generate the A/D conversion start request signal (n = 0, 1).
Figure 20.96 shows an example of outputting the A/D conversion start request frame synchronization signal.

Corrections

This function can be used to monitor the generation timing of the A/D conversion start request signal using an external pin. When the A/D conversion request signal to be monitored is selected by the TADSTRGR0 register, a pulse signal is output from the ADSM0 pin that is at the high level when the A/D conversion start request signal is generated, and at the low level in the timer cycle used to generate the A/D conversion start request signal.
Figure 20.99 shows an example of outputting the A/D conversion start request frame synchronization signal.

•Page 490, Table 20.63 MTU Interrupt Sources is corrected as follows:

Before correction

Channel	Name	Interrupt Source	DMAC/DTC Activation
MTU0	TGIA0	MTU0.TGRA input capture/compare match	Possible
	TGIB0	MTU0.TGRB input capture/compare match	Possible
	TGIC0	MTU0.TGRC input capture/compare match	Possible
	TGID0	MTU0.TGRD input capture/compare match	Possible
	TCIV0	MTU0.TCNT overflow	Not possible
	TGIE0	MTU0.TGRE compare match	Not possible
	TGIF0	MTU0.TGRF compare match	Not possible
MTU1	TGIA1	MTU1.TGRA input capture/compare match	Possible
	TGIB1	MTU1.TGRB input capture/compare match	Possible
	TCIV1	MTU1.TCNT overflow	Not possible
	TCIU1	MTU1.TCNT underflow	Not possible
MTU2	TGIA2	MTU2.TGRA input capture/compare match	Possible
	TGIB2	MTU2.TGRB input capture/compare match	Possible
	TCIV2	MTU2.TCNT overflow	Not possible
	TCIU2	MTU2.TCNT underflow	Not possible
MTU3	TGIA3	MTU3.TGRA input capture/compare match	Possible
	TGIB3	MTU3.TGRB input capture/compare match	Possible
	TGIC3	MTU3.TGRC input capture/compare match	Possible
	TGID3	MTU3.TGRD input capture/compare match	Possible
	TCIV3	MTU3.TCNT overflow	Not possible
MTU4	TGIA4	MTU4.TGRA input capture/compare match	Possible
	TGIB4	MTU4.TGRB input capture/compare match	Possible
	TGIC4	MTU4.TGRC input capture/compare match	Possible
	TGID4	MTU4.TGRD input capture/compare match	Possible
	TCIV4	MTU4.TCNT overflow/underflow*1	Possible
MTU5	TGIU5	MTU5.TGRU input capture/compare match	Possible
	TGIV5	MTU5.TGRV input capture/compare match	Possible
	TGIW5	MTU5.TGRW input capture/compare match	Possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.
Note 1. Only in complementary PWM mode

Corrections

Channel	Name	Interrupt Source	DTC Activation	Priority
MTU0	TGIA0	MTU0.TGRA input capture/compare match	Possible	High
	TGIB0	MTU0.TGRB input capture/compare match	Possible	
	TGIC0	MTU0.TGRC input capture/compare match	Possible	
	TGID0	MTU0.TGRD input capture/compare match	Possible	
	TCIV0	MTU0.TCNT overflow	Not possible	
	TGIE0	MTU0.TGRE compare match	Not possible	
	TGIF0	MTU0.TGRF compare match	Not possible	
MTU1	TGIA1	MTU1.TGRA input capture/compare match	Possible	
	TGIB1	MTU1.TGRB input capture/compare match	Possible	
	TCIV1	MTU1.TCNT overflow	Not possible	
	TCIU1	MTU1.TCNT underflow	Not possible	
MTU2	TGIA2	MTU2.TGRA input capture/compare match	Possible	
	TGIB2	MTU2.TGRB input capture/compare match	Possible	
	TCIV2	MTU2.TCNT overflow	Not possible	
	TCIU2	MTU2.TCNT underflow	Not possible	
MTU3	TGIA3	MTU3.TGRA input capture/compare match	Possible	
	TGIB3	MTU3.TGRB input capture/compare match	Possible	
	TGIC3	MTU3.TGRC input capture/compare match	Possible	
	TGID3	MTU3.TGRD input capture/compare match	Possible	
	TCIV3	MTU3.TCNT overflow	Not possible	
MTU4	TGIA4	MTU4.TGRA input capture/compare match	Possible	
	TGIB4	MTU4.TGRB input capture/compare match	Possible	
	TGIC4	MTU4.TGRC input capture/compare match	Possible	
	TGID4	MTU4.TGRD input capture/compare match	Possible	
	TCIV4	MTU4.TCNT overflow/underflow*1	Possible	
MTU5	TGIU5	MTU5.TGRU input capture/compare match	Possible	
	TGIV5	MTU5.TGRV input capture/compare match	Possible	
	TGIW5	MTU5.TGRW input capture/compare match	Possible	Low

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.
Note 1. Only in complementary PWM mode

•Page 491, (1) Input Capture/Compare Match Interrupt is corrected as follows:

Before correction

If the TIER.TGIE bit is set to 1 when a TGR input capture/compare match occurs on a channel, an interrupt is requested. The MTU has 21 input capture/compare match interrupts (six for MTU0, four each for MTU3 and MTU4, two each for MTU1 and MTU2, and three for MTU5). The MTU0.TGFE and MTU0.TGFF_0 flags in MTU0 are not set by the occurrence of an input capture.

Corrections

If the TIER.TGIE bit is set to 1 when a TGR input capture/compare match occurs on a channel, an interrupt is requested. The MTU has 21 input capture/compare match interrupts (six for MTU0, four each for MTU3 and MTU4, two each for MTU1 and MTU2, and three for MTU5).

•Page 491, (2) Overflow Interrupt is corrected as follows:

Before correction

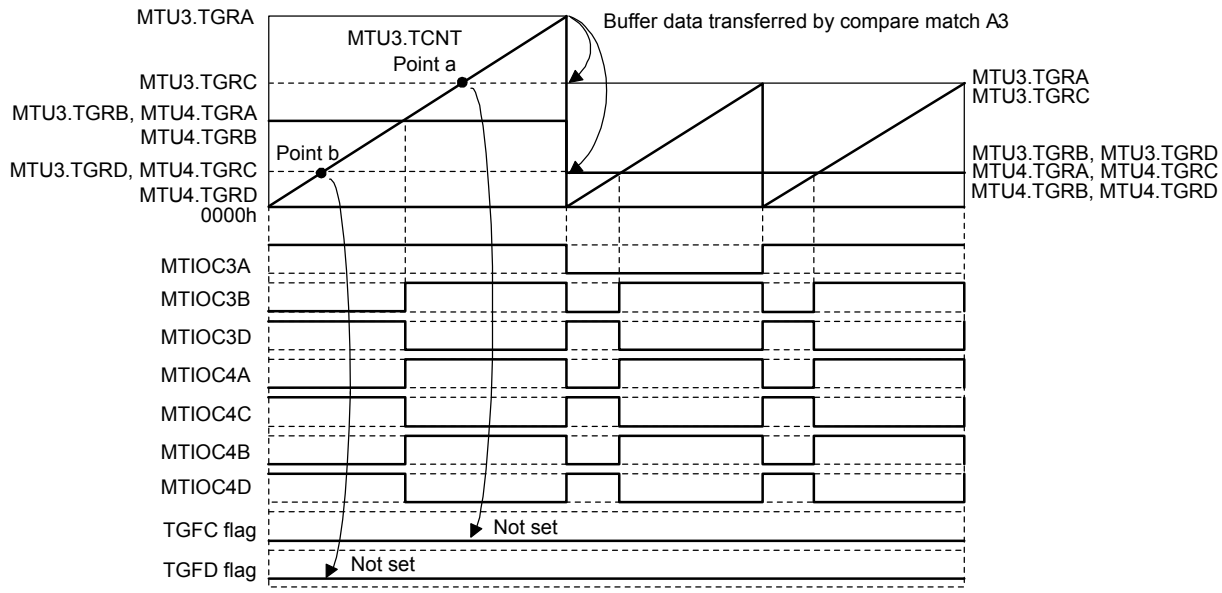
If the TIER.TCIEV bit is set to 1 when a TCNT overflow occurs on a channel, clearing an interrupt is requested. The MTU has five overflow interrupts (one for each channel).

Corrections

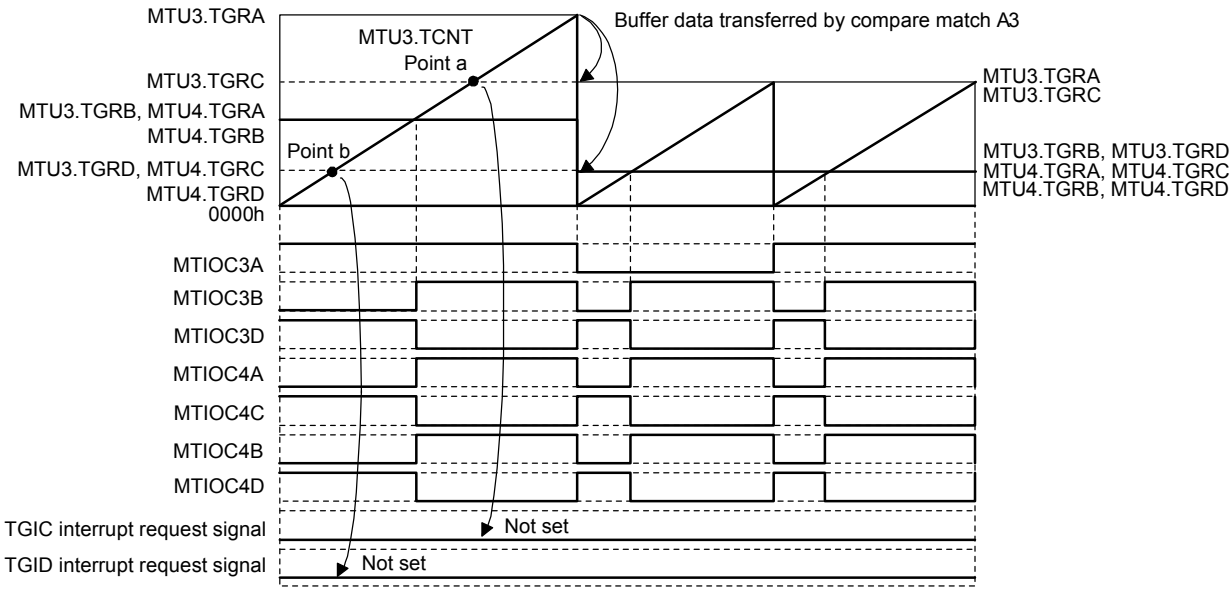
If the TIER.TCIEV bit is set to 1 when a TCNT overflow occurs on a channel, clearing an interrupt is requested. The MTU has five overflow interrupts (one for each channel except MTU5).

•Page 512, Figure 20.131 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode is corrected as follows:

Before correction



Corrections



•Page 519, “Usage Notes on A/D Converter Delaying Function in Complementary PWM Mode” is corrected as follows:

Before correction

- To issue an A/D converter start request linked with the interrupt skipping function, set the MTU4.TADCORA and MTU4.TADCORB registers so that 2 ≤ MTUn.TADCORA/TADCORB ≤ TCDR - 2 is satisfied (n = 4, 7).

Corrections

- To issue an A/D converter start request linked with the interrupt skipping function, set the MTU4.TADCORA and MTU4.TADCORB registers so that 2 ≤ MTUn.TADCORA/TADCORB ≤ TCDR - 2 is satisfied (n = 4).

•Page 553, “Input Level Control/Status Register 6 (ICSR6)” is corrected as follows:

Before correction

OSTSTF Flag (OSTST High-Impedance Flag)

This flag indicates that a oscillation stop high-impedance request has been generated.
When oscillation stop is detected, this flag is set to 1. When clearing this flag to 0, do so while the oscillation stop detection signal is negated; writing 0 to this flag while the oscillation stop detection signal is asserted does not clear this flag to 0.
After clearing this flag, confirm that the flag has actually been modified to 0.

Corrections

OSTSTF Flag (OSTST High-Impedance Flag)

This flag indicates that a oscillation stop high-impedance request has been generated.
When oscillation stop is detected, this flag is set to 1. To clear this flag, wait for at least 10 cycles of PCLKB after this flag becomes 1 and write 0 to this flag while the OSTDSR.OSTDF flag is 0. Writing 0 to this flag while the OSTDSR.OSTDF flag is 1 cannot clear this flag. After clearing this flag, confirm that the flag has actually been modified to 0.

•Page 583, “Timer Control Register (TCR)” is corrected as follows:

Before correction

Note 1. To use an external reset, set the PORTn.PDR.Bn bit for the corresponding pin to 0 and the PORTn.PMR.Bn bit to 1. For details, see section 18, I/O Ports.

Corrections

Note 1. To use an external counter reset signal, set the corresponding pin function. For details, refer to section 18, I/O Ports and section 19, Multi-Function Pin Controller (MPC).

•Page 584, “Timer Counter Control Register (TCCR)” is corrected as follows:

Before correction

Note 1. To use an external reset, set the PORTn.PDR.Bn bit for the corresponding pin to 0 and the PORTn.PMR.Bn bit to 1. For details, see section 18, I/O Ports.

Corrections

Note 1. To use an external count clock, set the corresponding pin function. For details, refer to section 18, I/O Ports and section 19, Multi-Function Pin Controller (MPC).

•Page 585, Table 22.5 Clock Input to TCNT and Count Condition is corrected as follows:

Before correction

Note 1. To use an external reset, set the PORTn.PDR.Bn bit for the corresponding pin to 0 and the PORTn.PMR.Bn bit to 1. For details, see section 18, I/O Ports.

Corrections

Note 1. To use an external count clock, set the corresponding pin function. For details, refer to section 18, I/O Ports and section 19, Multi-Function Pin Controller (MPC).

•Page 606, Table 23.2 CMT Interrupt Sources is corrected as follows:

Before correction

Name	Interrupt Sources	DTC Activation
CMI0	Compare match in CMT0	Possible
CMI1	Compare match in CMT1	Possible

Corrections

Name	Interrupt Sources	DTC Activation
CMI0	Compare match in CMT0	Possible
CMI1	Compare match in CMT1	Possible
CMI2	Compare match in CMT2	Possible
CMI3	Compare match in CMT3	Possible

•Page 608, “Overview” is corrected as follows:

Before correction

- When making a transition to sleep mode, software standby mode, or deep sleep mode, the IWDTCSTPR.SLCSTP bit can be used to select whether to stop the counter or not.

Corrections

- When making a transition to sleep mode, software standby mode, or deep sleep mode, the IWDTCSTPR.SLCSTP bit or the OFS0.IWDTSLCSTP bit can be used to select whether to stop the counter or not.

•Page 617, “Register Start Mode” is corrected as follows:

Before correction

Set the IWDT reset interrupt request select bit (OFS0.IWDTRSTIRQS) to select either reset output or interrupt request output.

Corrections

Set the IWDT reset interrupt request select bit (IWDTRCR.RSTIRQS) to select either reset output or interrupt request output.

•Page 622, “Refresh Operation” is corrected as follows:

Before correction

[Sample refreshing timings]

- When the window start position is set to 03FFh, even if 00h is written to IWDTRR before 1FFFh is reached (2002h, for example), refreshing is done if FFh is written to IWDTRR after the value of the IWDTSR.CNTVAL[13:0] bits has reached 1FFFh.
- When the window end position is set to 1FFFh, refreshing is done if 0403h (four-count cycles before 03FFh) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to IWDTRR.

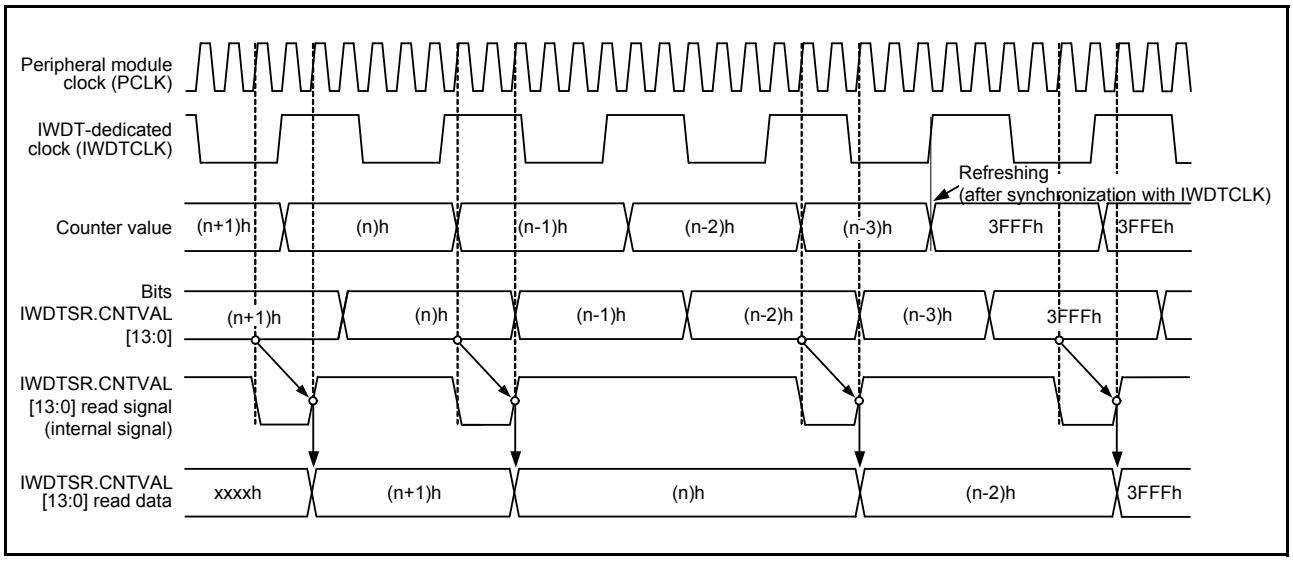
Corrections

[Sample refreshing timings]

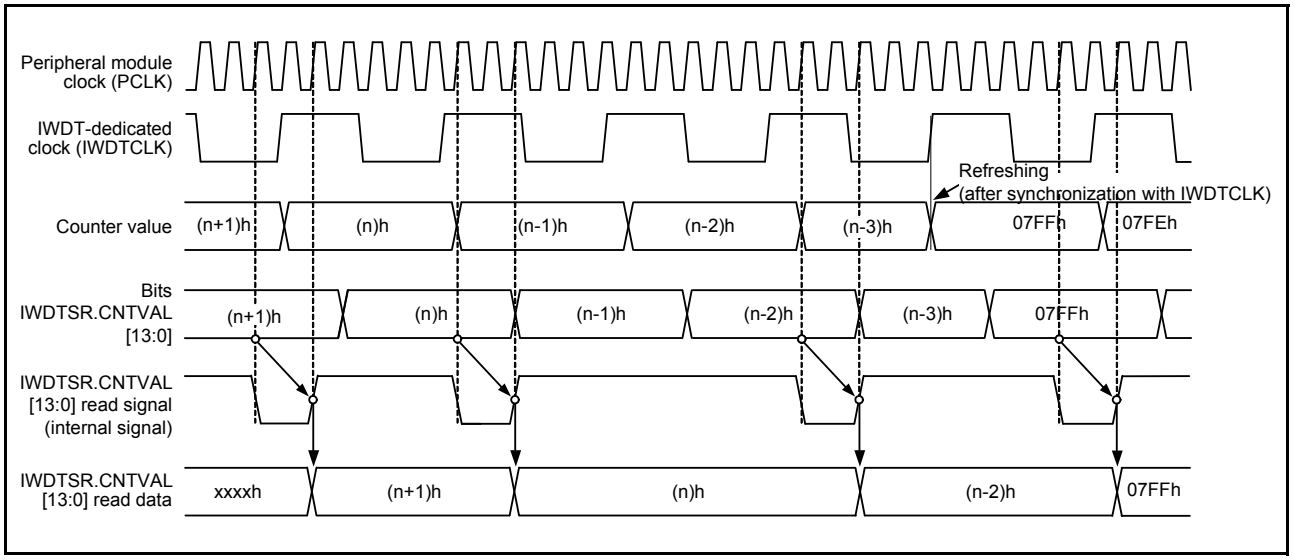
- When the window start position is set to 03FFh, even if 00h is written to the IWDTRR register before 03FFh is reached (0402h, for example), refreshing is done if FFh is written to the IWDTRR register after the value of the IWDTSR.CNTVAL[13:0] bits has reached 03FFh.
- When the window end position is set to 03FFh, refreshing is done if 0403h (four-count cycles before 03FFh) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to the IWDTRR register.

•Page 625, Figure 22.5 Processing for Reading IWDT Counter Value is corrected as follows:

Before correction



Corrections



•Page 640, “Serial Control Register (SCR)” is corrected as follows:

Before correction

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

Corrections

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that the SMR register should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, PER, and RDRF flags in the SSR register are not affected and the previous value is retained.

•Page 653, Table 25.13 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode) is corrected as follows:

Before correction

PCLK (MHz)	Maximum Bit Rate (bps)	n	N
8	250000	0	0
9.8304	307200	0	0
10	312500	0	0
12	375000	0	0
12.288	384000	0	0
14	437500	0	0
16	500000	0	0
17.2032	537600	0	0
18	562500	0	0
19.6608	614400	0	0
20	625000	0	0
25	781250	0	0
30	937500	0	0
33	1031250	0	0
40	1250000	0	0

Corrections

	SEMR Settings						SEMR Settings				
PCLK (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bps)	PCLK (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bps)
8	0	0	0	0	250000	18	0	0	0	0	562500
		1	0	0	500000			1	0	0	1125000
	1	0	0	0			1	0	0	0	
		1	0	0	1000000			1	0	0	2250000
9.8304	0	0	0	0	307200	19.6608	0	0	0	0	614400
		1	0	0	614400			1	0	0	1228800
	1	0	0	0			1	0	0	0	
		1	0	0	1228800			1	0	0	2457600
10	0	0	0	0	312500	20	0	0	0	0	625000
		1	0	0	625000			1	0	0	1250000
	1	0	0	0			1	0	0	0	
		1	0	0	1250000			1	0	0	2500000
12	0	0	0	0	375000	25	0	0	0	0	781250
		1	0	0	750000			1	0	0	1562500
	1	0	0	0			1	0	0	0	
		1	0	0	1500000			1	0	0	3125000
12.288	0	0	0	0	384000	30	0	0	0	0	937500
		1	0	0	768000			1	0	0	1875000
	1	0	0	0			1	0	0	0	
		1	0	0	1536000			1	0	0	3750000
14	0	0	0	0	437500	33	0	0	0	0	1031250
		1	0	0	875000			1	0	0	2062500
	1	0	0	0			1	0	0	0	
		1	0	0	1750000			1	0	0	4125000
16	0	0	0	0	500000	40	0	0	0	0	1250000
		1	0	0	1000000			1	0	0	2500000
	1	0	0	0			1	0	0	0	
		1	0	0	2000000			1	0	0	5000000
17.2032	0	0	0	0	537600						
		1	0	0	1075200						
	1	0	0	0							
		1	0	0	2150400						

•Page 658, “Serial Extended Mode Register (SEMR)” is corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description	R/W
b0	ACS0	Asynchronous Mode Clock Source Select	(Valid only in asynchronous mode) 0: External clock input 1: Logical AND of two clock cycles output from the TMR (valid only for SCI5) The following table lists the correspondence between SCI channels and compare match outputs.	R/W*1

SCI	TMR	Compare Match Output
SCI5	Unit 0	TMO0, TMO1

Corrections

Bit	Symbol	Bit Name	Description	R/W
b0	ACS0	Asynchronous Mode Clock Source Select	(Valid only in asynchronous mode) 0: External clock input 1: Logical AND of two compare matches output from TMR (valid for SCI5 only) Available compare match output varies per SCI channel.	R/W*1

Before correction

SEMR selects the clock source for 1-bit period in asynchronous mode.

For SCI5, the TMOn output (n = 0, 1) of TMR unit 0 can be set as the serial transfer base clock.

Figure 25.3 shows a setting example when the TMOn output (n = 0, 1) of TMR0 is selected.

Corrections

The SEMR register is used to select a clock source for 1-bit period in asynchronous mode or a detection method of the start bit.

Before correction

ACS0 Bit (Asynchronous Mode Clock Source Select)

Selects the clock source in the asynchronous mode.

The ACS0 bit is valid in asynchronous mode (CM bit in SMR = 0) and when an external clock input is selected (CKE[1:0] bits in SCR = 10b or 11b). An external clock input or internal TMR clock input can be selected.

Set the ACS0 bit to 0 in other than asynchronous mode.

This bit for the other SCI channel than SCI5 is reserved. The write values to this bit for other than SCI5 should be 0.

Corrections

ACS0 Bit (Asynchronous Mode Clock Source Select)

Selects the clock source in the asynchronous mode.

The ACS0 bit is valid in asynchronous mode (SMR.CM bit = 0) and when an external clock input is selected (SCR.CKE[1:0] bits = 10b or 11b). This bit is used to select an external clock input or the logical AND of compare matches output from the internal TMR.

Set the ACS0 bit to 0 in other than asynchronous mode.

For SCI5, the TMOn output (n = 0, 1) of TMR unit 0 can be set as the serial transfer base clock. Refer to Table 25.23 for details.

The ACS0 bit for SCI1 is reserved. The write value to this bit for SCI1 should be 0.

Table 25.23 Correspondence between SCI Channels and Compare Match Outputs

SCI	TMR	Compare Match Output
SCI5	Unit 0	TMO0, TMO1

Figure 25.23 shows a setting example of when TMO0 and TMO1 in the TMR unit 0 are selected for output.

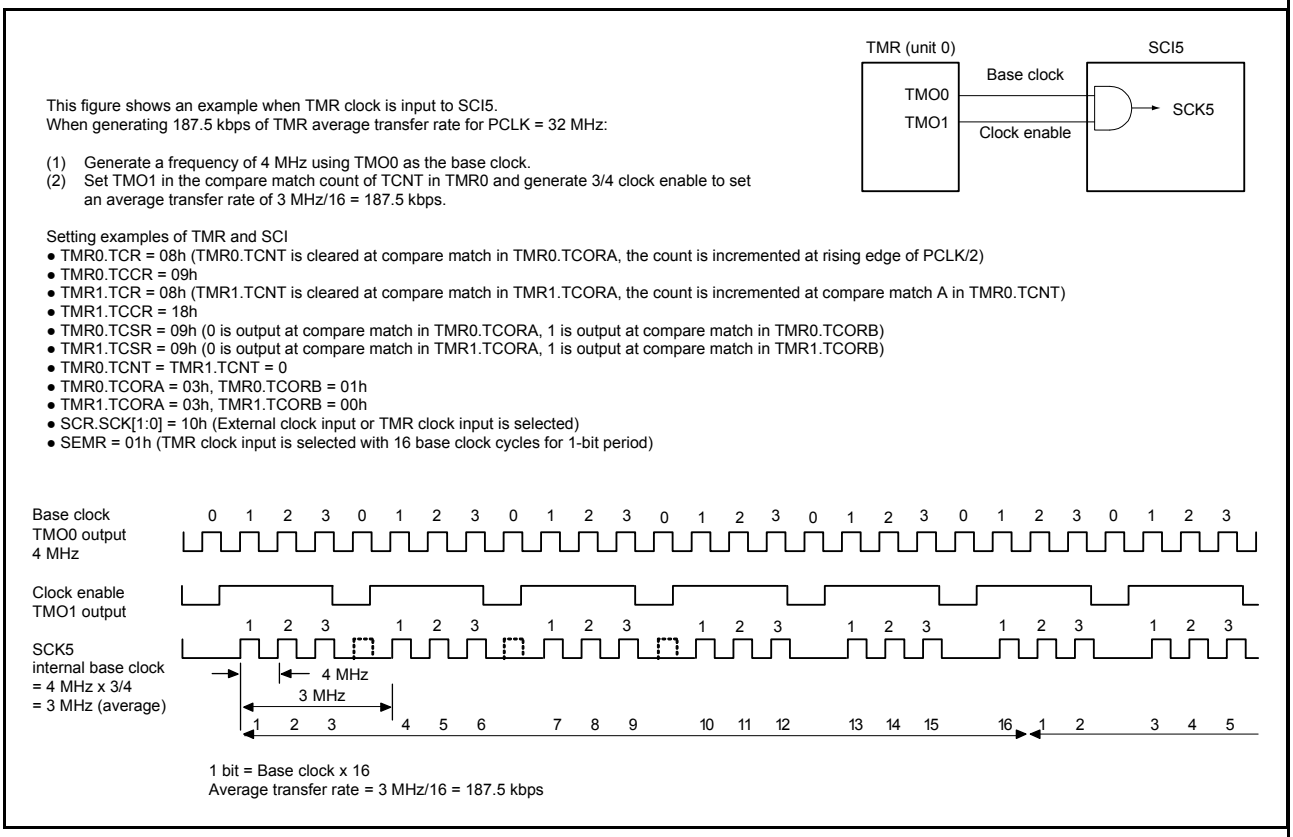


Figure 25.23 Example of Average Transfer Rate Setting When TMR Clock is Input

•Page 673, “SCI Initialization (Asynchronous Mode)” is corrected as follows:

Before correction

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization. Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR, RDRH, and RDRL.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.

Corrections

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization. Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, PER, and RDRF flags in the SSR register nor registers RDR, RDRH, and RDRL.

Moreover, note that changing the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a transmit data empty interrupt (TXI) request.

In addition, note that setting bits TIE, TE, and TEIE in the SCR register to 1 simultaneously leads to the generation of a transmit end interrupt (TEI) request before the generation of a TXI interrupt request.

•Page 724, Table 25.26 Interrupt Sources is corrected as follows:

Before correction

Name	**Interrupt Source**	Interrupt Flag	DTC Activation	Priority
ERI	Receive error	ORER, FER, or PER	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	—	Possible	
TEI	Transmit end	TEND	Not possible	Low

Corrections

Name	**Interrupt Source**	Interrupt Flag	DTC Activation	Priority
ERI	Receive error	ORER, FER, or PER	Not possible	High
RXI	Receive data full	RDRF	Possible	↑
TXI	Transmit data empty	TDRE	Possible	
TEI	Transmit end	TEND	Not possible	Low

•Page 756, "I2C Bus Status Register 1 (ICSR1)" is corrected as follows:

Before correction

HOA Flag (Host Address Detection Flag)

[Clearing conditions]

- When 0 is written to the HOA bit after reading HOA = 1
- When a stop condition is detected
- When 0 is written to the SMBS bit in ICMR3 or the HOAE bit in ICSER
- When the received slave address does not match the host address (0001 000b) with the HOAE bit in ICSER set to 1 (host address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Corrections

HOA Flag (Host Address Detection Flag)

[Clearing conditions]

- When 0 is written to the HOA flag after reading HOA flag to be 1
- When a stop condition is detected
- When the received slave address does not match the host address (0001 000b) with the ICSER.HOAE bit set to 1 (host address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the first byte.
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

•Page 818, “RSPI Status Register (SPSR)” is corrected as follows:

Before correction

Address(es): RSPI0.SPSR 0008 8383h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PERF	MODF	IDLNF	OVRF
Value after reset:	x	0	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	OVRF	Overrun Error Flag	0: No overrun error occurs 1: An overrun error occurs	R/(W) *1
b1	IDLNF	RSPI Idle Flag	0: RSPI is in the idle state 1: RSPI is in the transfer state	R
b2	MODF	Mode Fault Error Flag	0: No mode fault error occurs 1: A mode fault error occurs	R/(W) *1
b3	PERF	Parity Error Flag	0: No parity error occurs 1: A parity error occurs	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	—	Reserved	The read value is undefined. The write value should be 1.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	—	Reserved	The read value is undefined. The write value should be 1.	R/W

Note 1. Only 0 can be written to clear the flag after reading 1.

Corrections

Address(es): RSPI0.SPSR 0008 8383h

	b7	b6	b5	b4	b3	b2	b1	b0
	SPRF	—	SPTEF	—	PERF	MODF	IDLNF	OVRF
Value after reset:	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OVRF	Overrun Error Flag	0: No overrun error occurs 1: An overrun error occurs	R/(W) *1
b1	IDLNF	RSPI Idle Flag	0: RSPI is in the idle state 1: RSPI is in the transfer state	R
b2	MODF	Mode Fault Error Flag	0: No mode fault error occurs 1: A mode fault error occurs	R/(W) *1
b3	PERF	Parity Error Flag	0: No parity error occurs 1: A parity error occurs	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	SPTEF	Transmit Buffer Empty Flag	0: Transmit buffer has valid data 1: Transmit buffer has no valid data	R/(W) *2
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	SPRF	Receive Buffer Full Flag	0: Receive buffer has no valid data 1: Receive buffer has valid data	R/(W) *2

Note 1. Only 0 can be written to clear the flag after reading 1.
Note 2. The write value should be 1.

•Page 819, the descriptions of the SPTEF and SPRF flags are added to “RSPI Status Register (SPSR)”.

SPTEF Flag (Transmit Buffer Empty Flag)

Indicates whether the transmit buffer (SPTX) in the RSPI data register has valid data.

[Setting condition]

- When the SPCR.SPE bit is 0 (disables the RSPI function)
- When data is transferred from the transmit buffer to the shift register

[Clearing condition]

- When the number of frames of transmit data specified by the SPDCR.SPFC[1:0] bits is written to the SPDR register

The SPDR register can be set only when the SPTEF flag is 1. The data in the transmit buffer is not updated when the SPDR register is set while the SPTEF flag is 0.

SPRF Flag (Receive Buffer Full Flag)

Indicates whether the receive buffer (SPRX) in the RSPI data register has valid data.

[Setting condition]

- When the number of frames of receive data specified by the SPDCR.SPFC[1:0] bits is transferred from shift register to the receive buffer (SPRX) while the SPCR.TXMD bit is 0 (full duplex) and the SPRF flag is 0.
  Note that the SPRF flag does not become 1 when the OVRF flag is 1.

[Clearing condition]

- When all of the received data are read from the SPDR register

•Page 822, “RSPI Data Register (SPDR)” is corrected as follows:

Before correction

The transmit buffer read pointer is updated when writing to SPDR, and not updated when reading from the transmit buffer. When reading from the transmit buffer, the value most recently written to SPDR is read. However, after generation of the transmit buffer empty interrupt, the values read from the transmit buffer are all 0 in the interval after completion of writing the number of frames of data specified in the number of frames specification bits (SPDCR.SPFC[1:0]) and before generation of the next buffer empty interrupt.

Corrections

The transmit buffer read pointer is updated when writing to SPDR, and not updated when reading from the transmit buffer. When reading from the transmit buffer, the value most recently written to SPDR is read. However, after generation of the transmit buffer empty interrupt, the values read from the transmit buffer are all 0 in the interval after completion of writing the number of frames of data specified in the number of frames specification bits (SPDCR.SPFC[1:0]) and before generation of the next buffer empty interrupt (while the SPSR.SPTEF flag is 0).

•Page 826, “RSPI Data Control Register (SPDCR)” is corrected as follows:

Before correction

SPFC[1:0] Bits (Number of Frames Specification)

The SPFC[1:0] bits specify the number of frames that can be stored in SPDR (per transfer activation). Up to four frames can be transmitted or received in one round of transmission or reception, and the amount of data is determined by the combination of the SPSCR.SPSLN[2:0] bits, and the SPDCR.SPFC[1:0] bits. Furthermore, the setting of the SPFC[1:0] bits adjusts the number of frames for generation of RSPI receive buffer full interrupt, and start of transmission or generation of transmit buffer empty interrupts. Table 27.4 lists the frame configurations that can be stored in SPDR and examples of combinations of settings for transmission and reception. If combinations of settings other than those shown in the examples are made, subsequent operations should not be performed.

Corrections

SPFC[1:0] Bits (Number of Frames Specification)

The SPFC[1:0] bits specify the number of frames that can be stored in SPDR (per transfer activation). Up to four frames can be transmitted or received in one round of transmission or reception, and the amount of data is determined by the combination of the SPSCR.SPSLN[2:0] bits, and the SPDCR.SPFC[1:0] bits. Furthermore, the setting of the SPFC[1:0] bits adjusts the number of frames for generation of RSPI receive buffer full interrupt, and start of transmission or generation of transmit buffer empty interrupts.

When the number of frames of transmit data specified by SPFC[1:0] bits is written to the SPDR register, the SPSR.SPTEF flag becomes 0 and transmission starts. Then, when the specified number of frames of transmit data has been transferred to the shift register, the SPTEF flag becomes 1 and the RSPI transmit buffer empty interrupt is generated.

When the number of frames specified by the SPFC[1:0] bits are received, the SPSR.SPRF flag becomes 1 and the RSPI receive buffer full interrupt is generated.

Table 27.4 lists the frame configurations that can be stored in SPDR and examples of combinations of settings for transmission and reception. If combinations of settings other than those shown in the examples are made, subsequent operations should not be performed.

•Page 827, “RSPI Data Control Register (SPDCR)” is corrected as follows:

Before correction

SPRDTD Bit (RSPI Receive/Transmit Data Select)

The SPRDTD bit selects whether the SPDR reads values from the receive buffer or from the transmit buffer.

If reading is from the transmit buffer, the value written to SPDR register immediately beforehand is read.

When reading the transmit buffer, do so before writing of the number of frames set in the SPFC[1:0] bits is finished and after generation of the transmit buffer empty interrupt.

For details, refer to section 27.2.5, RSPI Data Register (SPDR).

Corrections

SPRDTD Bit (RSPI Receive/Transmit Data Select)

The SPRDTD bit selects whether the SPDR reads values from the receive buffer or from the transmit buffer.

If reading is from the transmit buffer, the value written to SPDR register immediately beforehand is read.

When reading the transmit buffer, do so before writing of the number of frames set in the SPFC[1:0] bits is finished and after generation of the transmit buffer empty interrupt (While the SPSR.SPTEF flag is 1).

For details, refer to section 27.2.5, RSPI Data Register (SPDR).

•Page 834, “RSPI Command Registers 0 to 7 (SPCMD0 to SPCMD7)” is corrected as follows:

Before correction

SPB[3:0] Bits (RSPI Data Length Setting)

The SPB[3:0] bits set a transfer data length for the RSPI in master mode or slave mode.

Corrections

SPB[3:0] Bits (RSPI Data Length Setting)

The SPB[3:0] bits set a transfer data length for the RSPI in master mode or slave mode. When the SPLW bit is 0, set the SPB[3:0] bits to “0100b” (8 bits) to “1111b” (16 bits).

•Page 836, Table 27.5 Relationship between RSPI Modes and SPCR Settings and Description of Each Mode is corrected as follows:

Before correction

Transfer activation method

Transmit buffer is written to at generation of a transmit buffer empty interrupt request

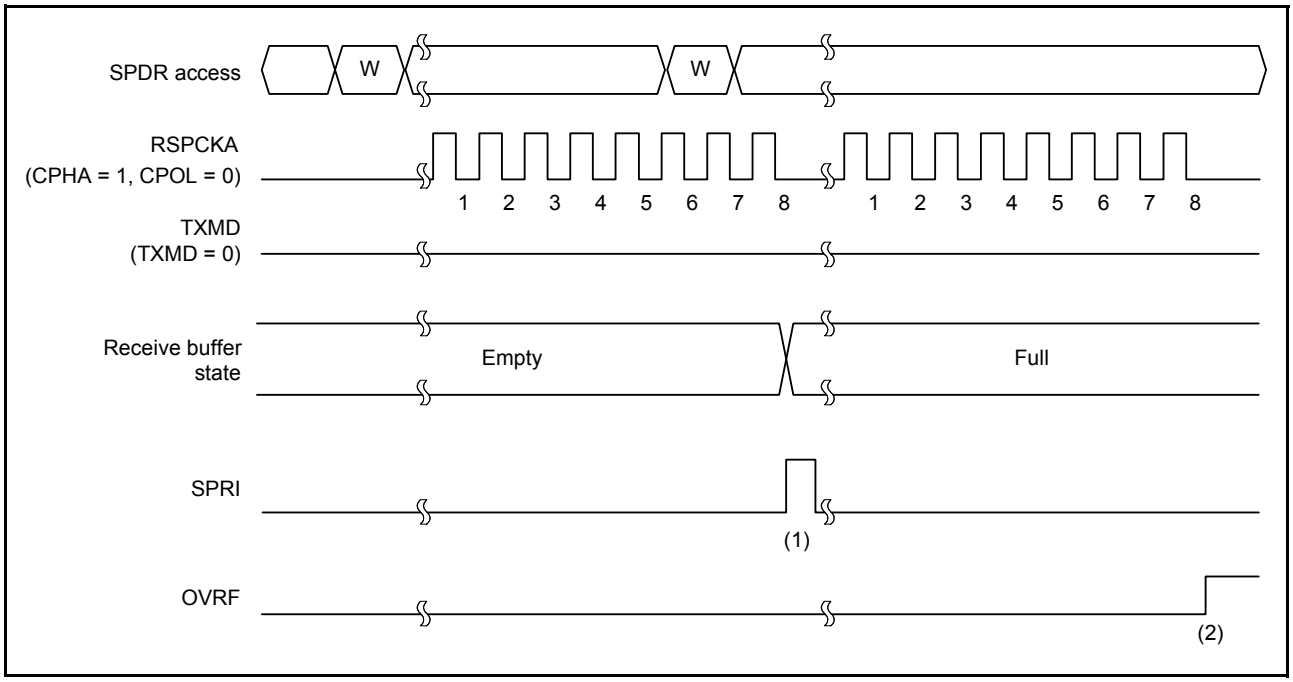
Corrections

Transfer activation method

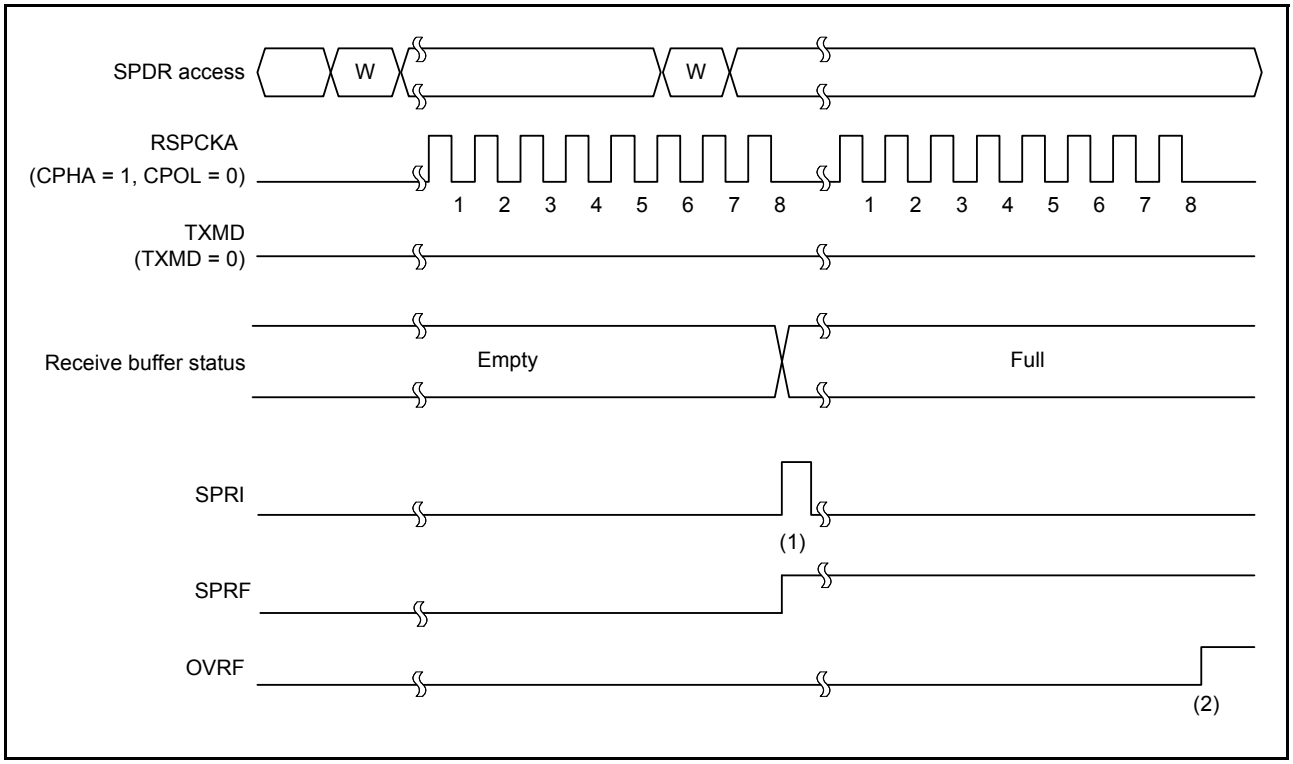
Transmit buffer is written to at generation of a transmit buffer empty interrupt request or when the SPTEF flag is 1

•Page 855, Figure 27.24 Operation Example of SPCR.TXMD = 0 is corrected as follows:

Before correction



Corrections



•Page 855, “Full-Duplex Synchronous Serial Communications (SPCR.TXMD = 0)” is corrected as follows:

Before correction

The operation of the flags at timings shown in steps (1) and (2) in the figure is described below.

(1) When a serial transfer ends with the receive buffer of SPDR empty, the RSPI generates a receive buffer full interrupt request (SPRI) and copies the received data in the shift register to the receive buffer.

(2) When a serial transfer ends with the receive buffer of SPDR holding data that was received in the previous serial transfer, the RSPI sets the SPSR.OVRF flag to 1 and discards the received data in the shift register.

Corrections

The operation of the flags at timings shown in steps (1) and (2) in the figure is described below.

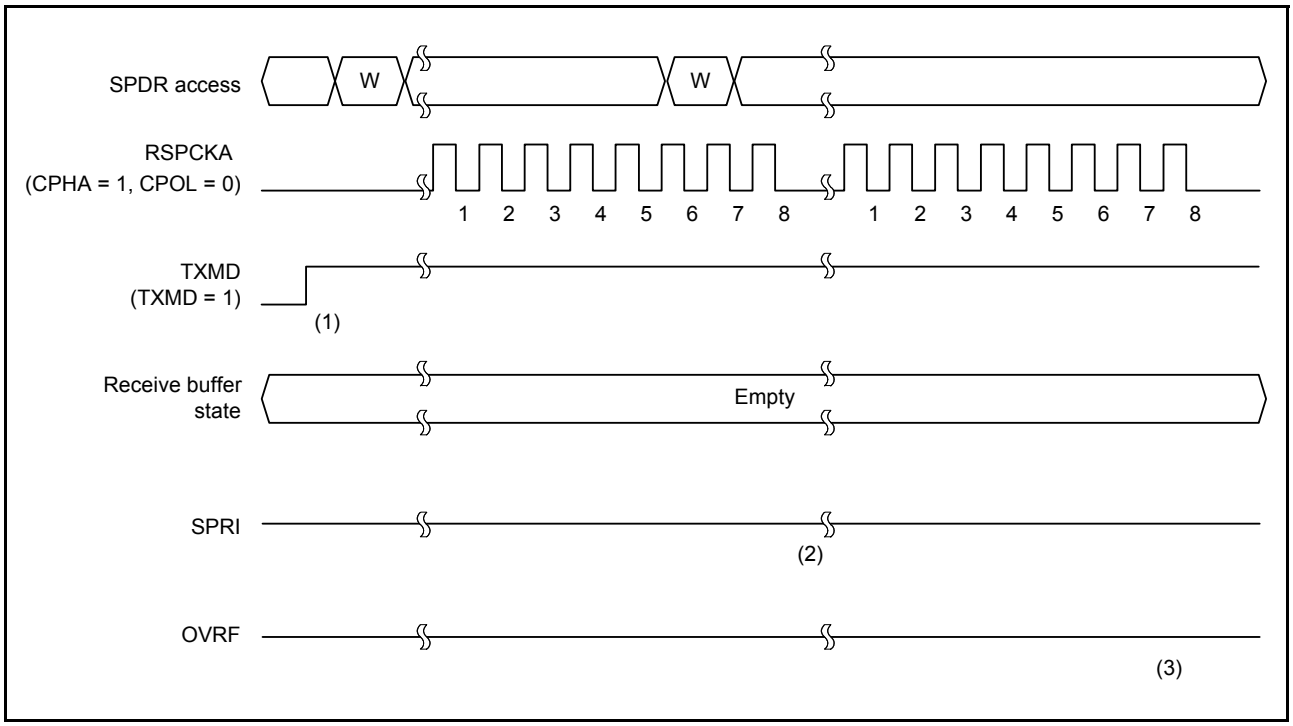
(1) When a serial transfer ends with the receive buffer of SPDR empty, the RSPI generates a receive buffer full interrupt request (SPRI) (sets the SPSR.SPRF flag to 1) and copies the received data in the shift register to the receive buffer.

(2) When a serial transfer ends with the receive buffer of SPDR holding data that was received in the previous serial transfer, the RSPI sets the SPSR.OVRF flag to 1 and discards the received data in the shift register.

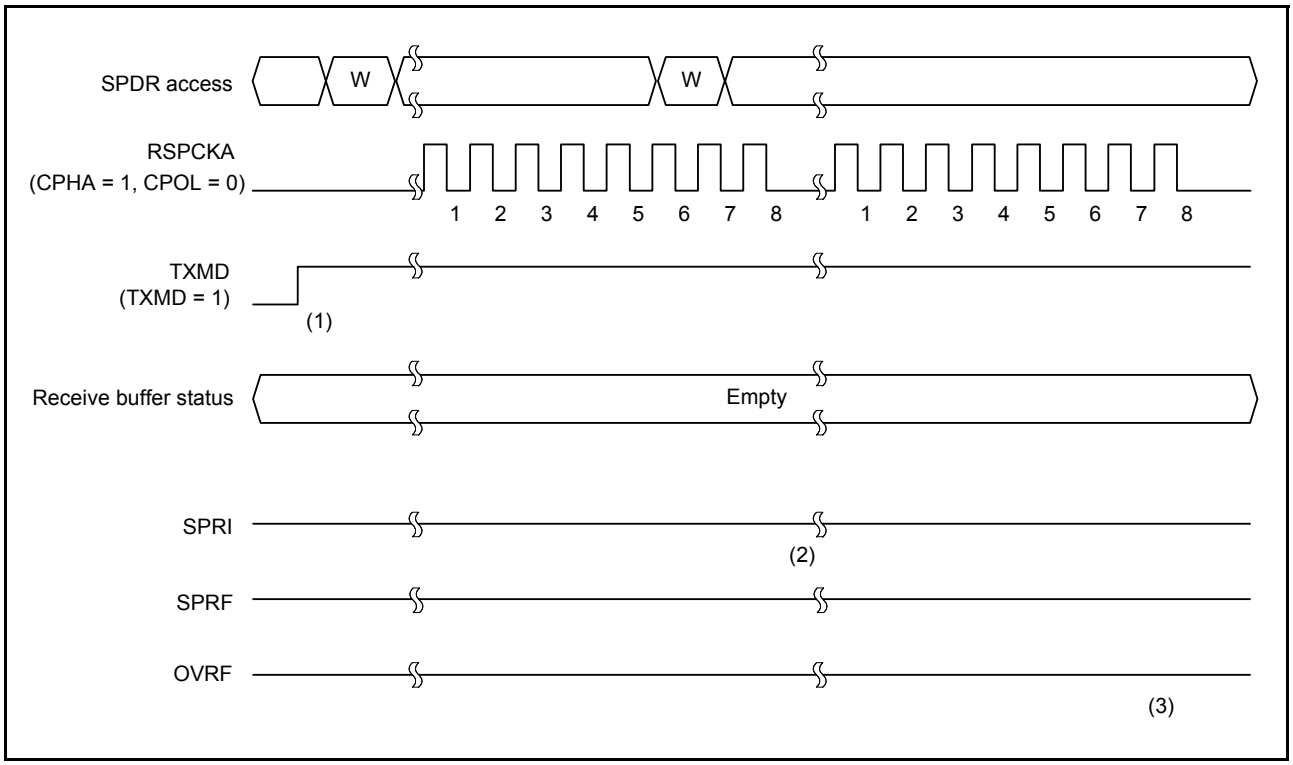
When full-duplex synchronous serial communications (SPCR.TXMD = 0) is selected, reception occurs simultaneously with transmit operations. As such, the SPRF and OVRF flags in the SPSR register become 1 at the timing described in (1) and (2), respectively, according to the state of the receive buffer.

•Page 856, Figure 27.25 Operation Example of SPCR.TXMD = 1 is corrected as follows:

Before correction



Corrections



•Page 856, “Transmit Operations Only (SPCR.TXMD = 1)” is corrected as follows:

Before correction

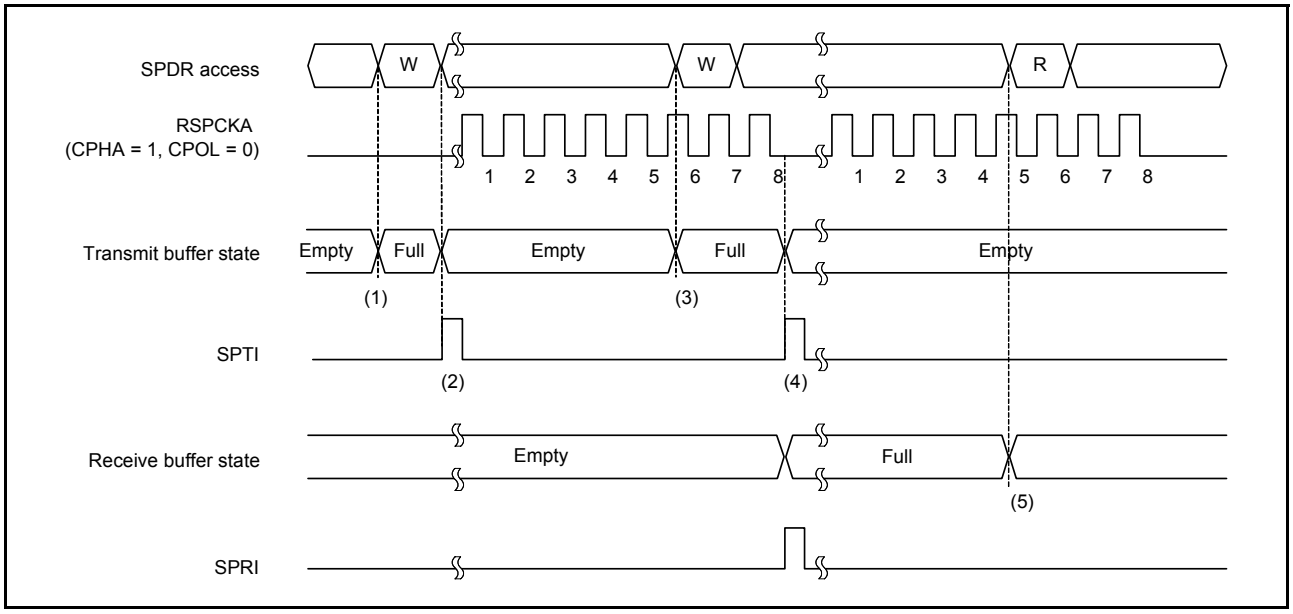
When performing transmit operations only (SPCR.TXMD = 1), the RSPI transmits transmit data but does not receive received data. Therefore, the SPSR.OVRF flag remains 0 at the timings of (1) to (3).

Corrections

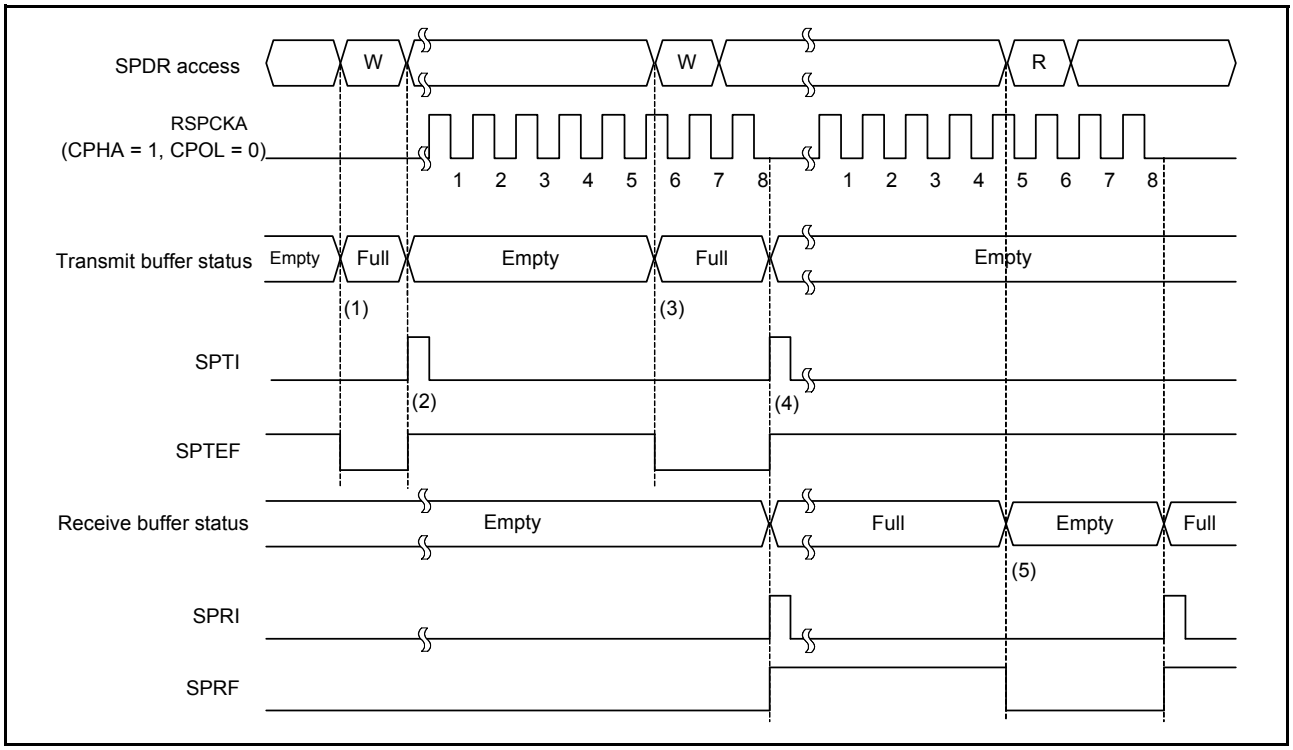
When performing transmit operations only (SPCR.TXMD = 1), the RSPI transmits data but does not receive data. Therefore, the SPSR.SPRF, OVRF flags remain 0 at the timings of (1) to (3).

•Page 857, Figure 27.26 Operation Example of SPTI and SPRI Interrupts is corrected as follows:

Before correction



Corrections



•Page 857, “Transmit Buffer Empty/Receive Buffer Full Interrupts” is corrected as follows:

Before correction

The operation of the interrupts at timings shown in steps (1) to (5) in the figure is described below.

1. When transmit data is written to SPDR when the transmit buffer of SPDR is empty (data for the next transfer is not set), the RSPI writes data to the transmit buffer.
2. If the shift register is empty, the RSPI copies the data in the transmit buffer to the shift register and generates a transmit buffer empty interrupt request (SPTI). How a serial transfer is started depends on the mode of the RSPI. For details, refer to section 27.3.10, SPI Operation, and section 27.3.11, Clock Synchronous Operation.
3. When transmit data is written to SPDR by the transmit buffer empty interrupt routine, the data is transferred to the transmit buffer. Because the data being transferred serially is stored in the shift register, the RSPI does not copy the data in the transmit buffer to the shift register.
4. When the serial transfer ends with the receive buffer of SPDR being empty, the RSPI copies the receive data in the shift register to the receive buffer and generates a receive buffer full interrupt request (SPRI). Since the shift register becomes empty upon completion of serial transfer, when the transmit buffer had been full before the serial transfer ended, the RSPI copies the data in the transmit buffer to the shift register. Even when received data is not copied from the shift register to the receive buffer in an overrun error status, upon completion of the serial transfer, the RSPI determines that the shift register is empty, thus data transfer from the transmit buffer to the shift register is enabled.
5. When SPDR is read by the receive buffer full interrupt routine, the receive data can be read.

Corrections

The operation of the interrupts at timings shown in steps (1) to (5) in the figure is described below.

(1) When transmit data is written to SPDR when the transmit buffer of SPDR is empty (data for the next transfer is not set), the RSPI writes data to the transmit buffer and sets the SPSR.SPTEF flag to 0.

(2) If the shift register is empty, the RSPI copies the data from the transmit buffer to the shift register and generates a transmit buffer empty interrupt request (SPTI) and sets the SPSR.SPTEF flag to 1. How a serial transfer is started depends on the mode of the RSPI. For details, refer to section 27.3.10, SPI Operation, and section 27.3.11, Clock Synchronous Operation.

(3) When transmit data is written to SPDR in the transmit buffer empty interrupt routine or in the transmit buffer empty detecting process by polling the SPTEF flag, the data is transferred to the transmit buffer and the SPSR.SPTEF flag becomes 0. Because the data being transmitted is stored in the shift register, the RSPI does not copy the data from the transmit buffer to the shift register.

(4) When the serial transfer ends with the receive buffer of SPDR being empty, the RSPI copies the receive data from the shift register to the receive buffer, generates a receive buffer full interrupt request (SPRI), and sets the SPSR.SPRF flag to 1. Since the shift register becomes empty upon completion of serial transfer, when the transmit buffer had been full before the serial transfer ended, the RSPI sets the SPSR.SPTEF flag to 1 and copies the data from the transmit buffer to the shift register. Even when received data is not copied from the shift register to the receive buffer in an overrun error status, upon completion of the serial transfer, the RSPI determines that the shift register is empty, thus data transfer from the transmit buffer to the shift register is enabled.

(5) When SPDR is read in the receive buffer full interrupt routine or in the receive buffer full detecting process by polling the SPRF flag, the receive data can be read. When the receive data is read, the SPRF flag becomes 0.

•Page 859, Table 27.8 Relationship between Non-Normal Transfer Operations and RSPI Error Detection Function is corrected as follows:

Before correction

	Occurrence Condition	RSPI Operation	Error Detection
1	SPDR is written when the transmit buffer is full.	• The contents of the transmit buffer are kept. • Missing write data.	None
2	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is transmitted.	None
3	SPDR is read when the receive buffer is empty.	Previously received data is output.	None
4	Serial transfer terminates when the receive buffer is full.	• The contents of the receive buffer are kept. • Missing receive data.	Overrun error
5	An incorrect parity bit is received when performing full-duplex synchronous serial communications with the parity function enabled.	The parity error flag is asserted.	Parity error
6	The SSLA0 input signal is asserted when the serial transfer is idle in multi-master mode.	• Driving of the RSPCKA, MOSIA, SSLA1 to SSLA3 output signals is stopped. • RSPI function is disabled.	Mode fault error
7	The SSLA0 input signal is asserted during serial transfer in multi-master mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the RSPCKA, MOSIA, SSLA1 to SSLA3 output signals is stopped. • RSPI function is disabled.	Mode fault error
8	The SSLA0 input signal is negated during serial transfer in slave mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the MISOA output signal is stopped. • RSPI function is disabled.	Mode fault error

Corrections

	Occurrence Condition	RSPI Operation	Error Detection
1	SPDR is written when the transmit buffer is full.	• The contents of the transmit buffer are kept. • Missing write data.	None
2	SPDR is read when the receive buffer is empty.	Data received previously is output to the bus.	None
3	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is transmitted.	None
4	Serial transfer terminates when the receive buffer is full.	• The contents of the receive buffer are kept. • Missing receive data.	Overrun error
5	An incorrect parity bit is received when performing full-duplex synchronous serial communications with the parity function enabled.	The parity error flag is asserted.	Parity error
6	The SSLA0 input signal is asserted when the serial transfer is idle in multi-master mode.	• Driving of the RSPCKA, MOSIA, SSLA1 to SSLA3 output signals is stopped. • RSPI function is disabled.	Mode fault error
7	The SSLA0 input signal is asserted during serial transfer in multi-master mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the RSPCKA, MOSIA, SSLA1 to SSLA3 output signals is stopped. • RSPI function is disabled.	Mode fault error
8	The SSLA0 input signal is negated during serial transfer in slave mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the MISOA output signal is stopped. • RSPI function is disabled.	Mode fault error

•Page 859, “Error Detection” is corrected as follows:

Before correction

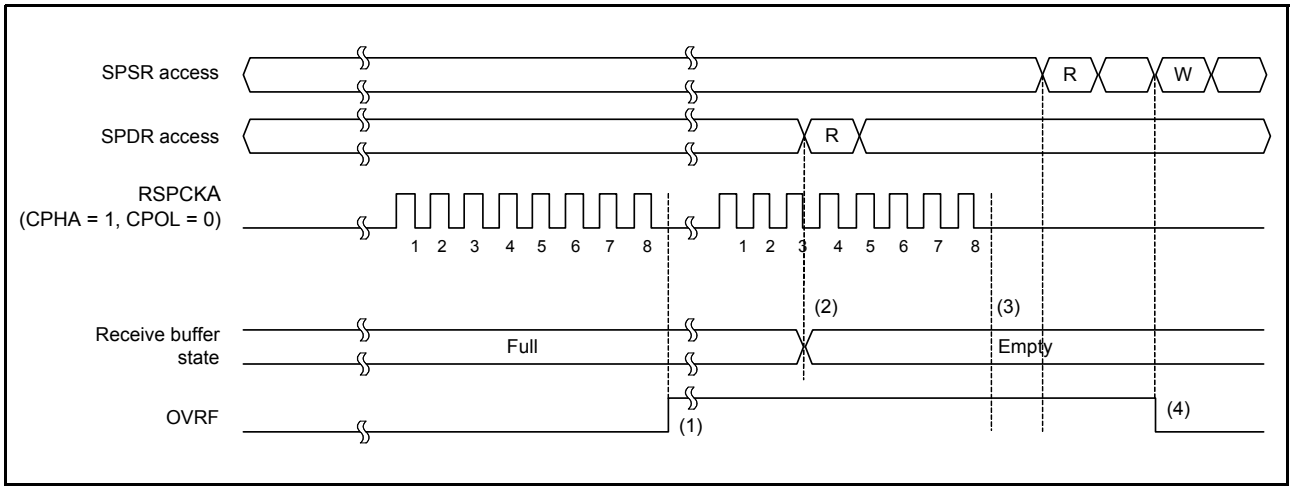
On operation 1 described in Table 27.8, the RSPI does not detect an error. To prevent data omission during the writing to SPDR, write operations to SPDR should be executed using a transmit buffer empty interrupt request.
Likewise, the RSPI does not detect an error on operation 2. In a serial transfer that was started before the shift register was updated, the RSPI sends the data that was received in the previous serial transfer, and does not treat the operation indicated in 2 as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read (if SPDR is not read before the end of the serial transfer, an overrun error may occur).
Similarly, the RSPI does not detect an error on operation 3. To prevent extraneous data from being read, SPDR read operation should be executed using an RSPI receive buffer full interrupt request.

Corrections

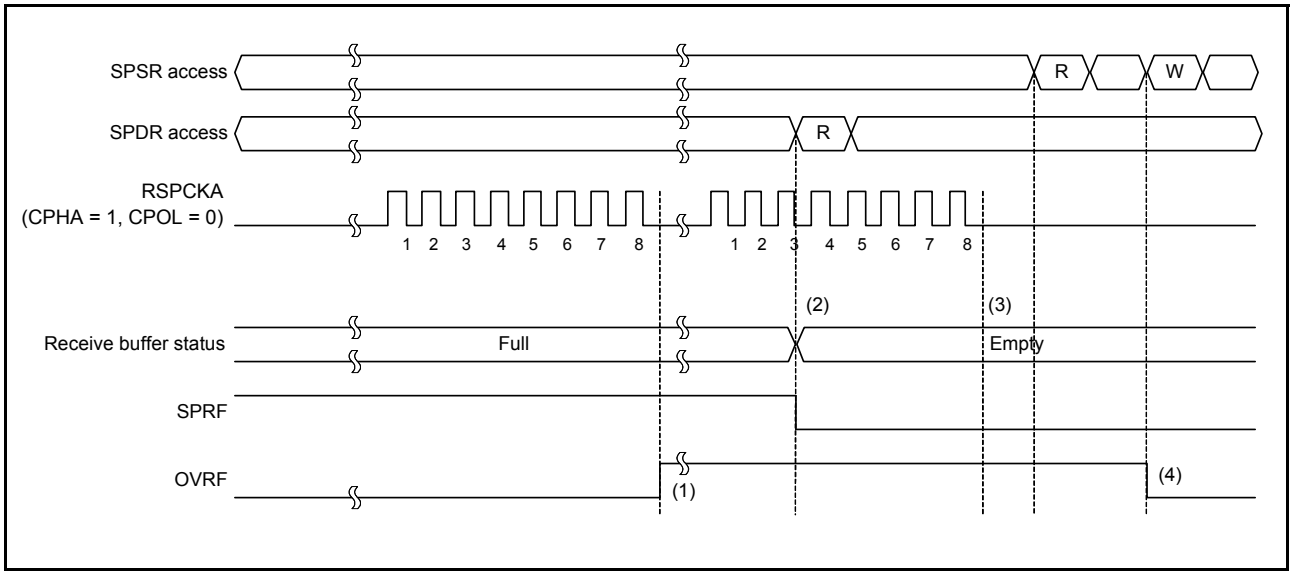
On operation 1 described in Table 27.8, the RSPI does not detect an error. To prevent data omission during the writing to SPDR, the SPDR register should be written when a transmit buffer empty interrupt request occurs or while the SPSR.SPTEF flag is 1.
Likewise, the RSPI does not detect an error on operation 2. To prevent extraneous data from being read, the SPDR register should be read when an RSPI receive buffer full interrupt request occurs or while the SPSR.SPRF flag is 1.
Similarly, the RSPI does not detect an error on operation 3. In a serial transfer that was started before the shift register was updated, the RSPI sends the data that was received in the previous serial transfer, and does not treat the operation indicated in 3 as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read (if SPDR is not read before the end of the serial transfer, an overrun error may occur).

•Page 860, Figure 27.27 Operation Example of OVRF Flag is corrected as follows:

Before correction



Corrections



•Page 860, “Overrun Error” is corrected as follows:

Before correction

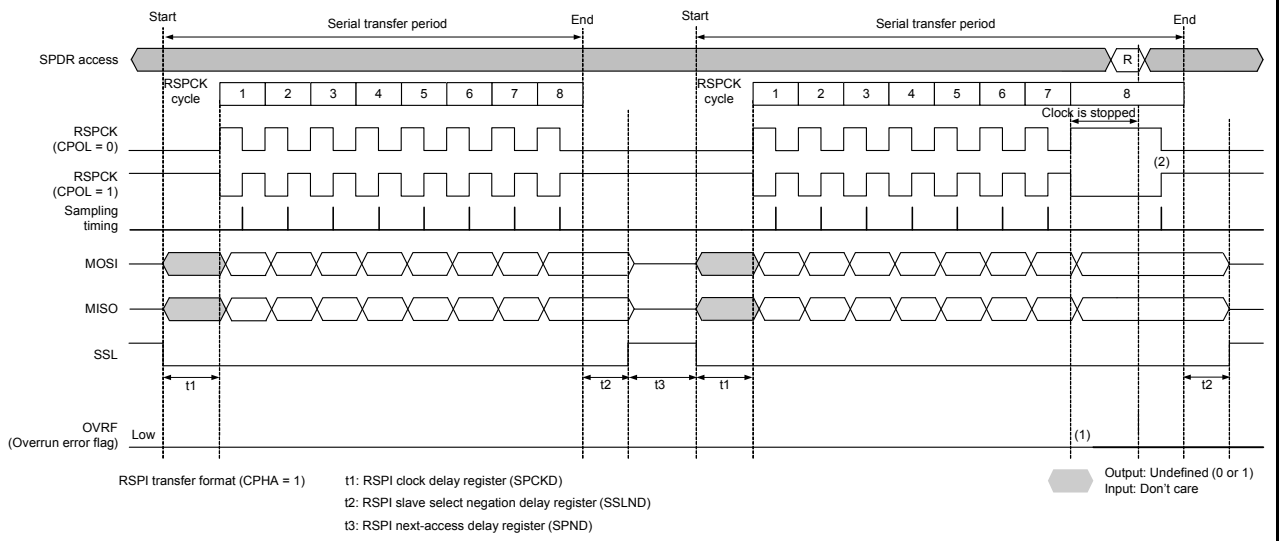
3. If the serial transfer ends with the OVRF flag being 1 (an overrun error occurs), the RSPI does not copy the data in the shift register to the receive buffer. A receive buffer full interrupt is not generated. Even if the SPPE bit is 1, parity errors are not detected. When in master mode, the RSPI does not update the SPSSR.SPECM[2:0] bits. When in an overrun error state and the RSPI does not copy the received data from the shift register to the receive buffer, upon termination of the serial transfer, the RSPI determines that the shift register is empty; in this manner, data transfer from the transmit buffer to the shift register is enabled.

Corrections

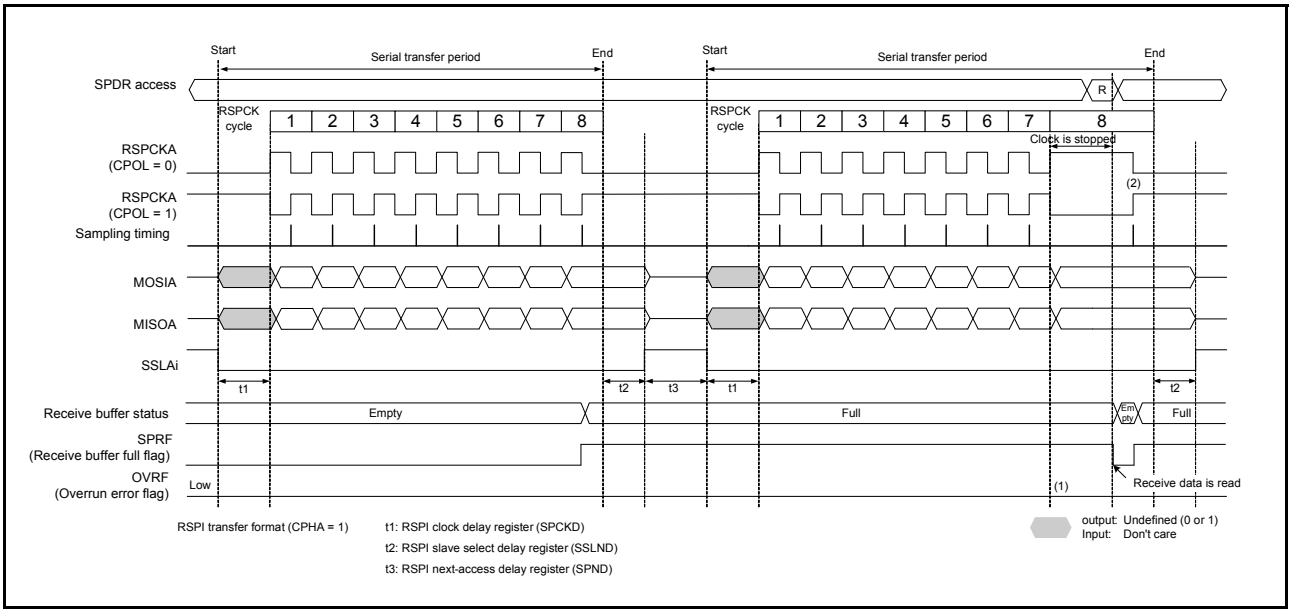
(3) If the serial transfer ends with the OVRF flag being 1 (an overrun error occurs), the RSPI does not copy the data in the shift register to the receive buffer (the SPRF flag remains 0). A receive buffer full interrupt is not generated. Even if the SPPE bit is 1, parity errors are not detected. When in master mode, the RSPI does not update the SPSSR.SPECM[2:0] bits. When in an overrun error state and the RSPI does not copy the received data from the shift register to the receive buffer, upon termination of the serial transfer, the RSPI determines that the shift register is empty; in this manner, data transfer from the transmit buffer to the shift register is enabled.

•Page 861, Figure 27.28 Clock Stop Waveform When a Serial Transfer Continues While the Receive Buffer is Full in Master Mode (CPHA = 1) is corrected as follows:

Before correction

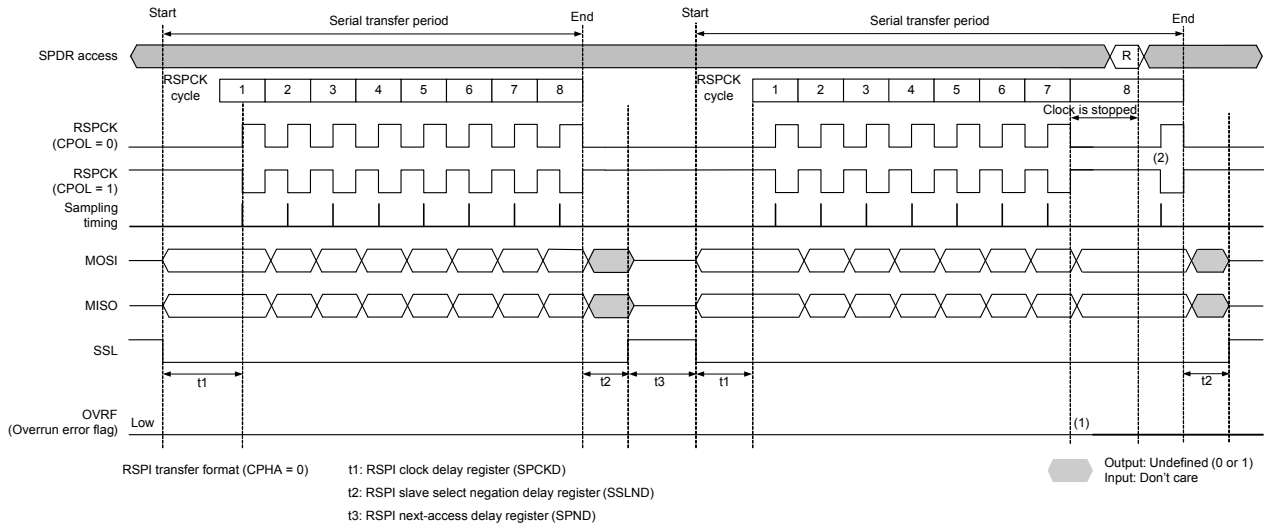


Corrections

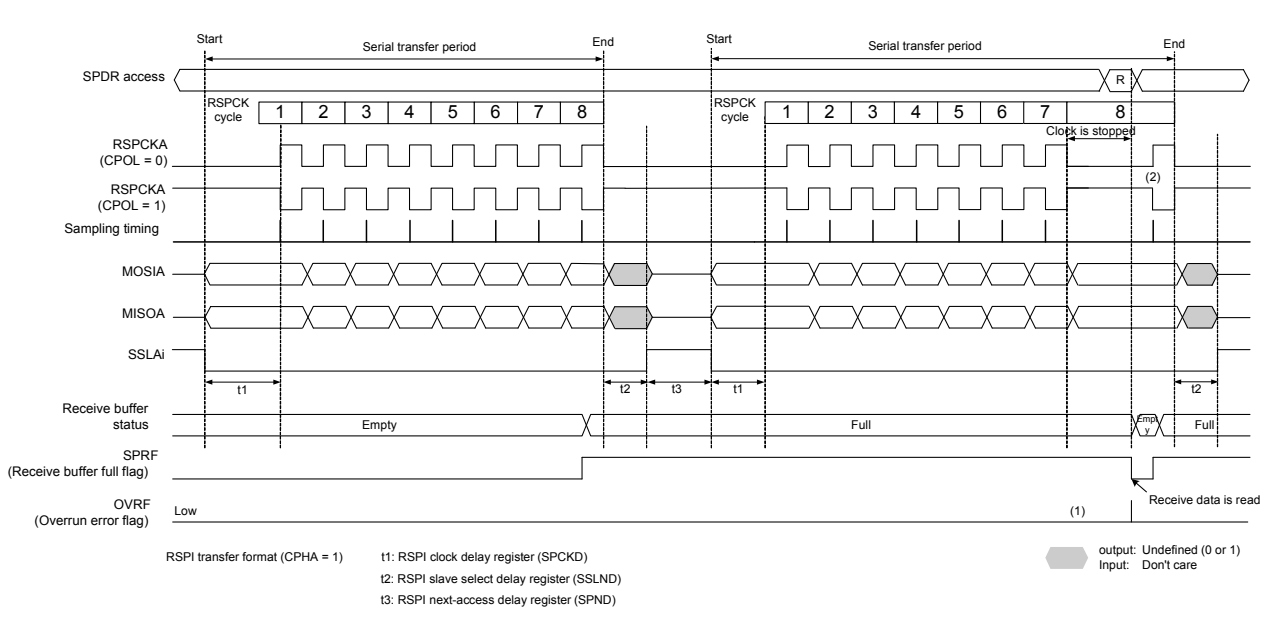


Page 861, Figure 27.29 Clock Stop Waveform When a Serial Transfer Continues While the Receive Buffer is Full in Master Mode (CPHA = 0) is corrected as follows:

Before correction



Corrections



•Page 864, “Initialization by Clearing the SPE Bit” is corrected as follows:

Before correction

- Initializing the transmit buffer of the RSPI

Corrections

- Initializing the transmit buffer of the RSPI (Set the SPTEF flag to 1)

Before correction

The SPSR.OVRF and SPSR.MODF flags are not initialized, nor is the value of the RSPI sequence status register (SPSSR) initialized. For this reason, even after the RSPI is initialized, data from the receive buffer can be read in order to check the status of error occurrence during an RSPI transfer.
The transmit buffer is initialized to an empty state. Therefore, if the SPCR.SPTIE bit is set to 1 after RSPI initialization, a transmit buffer empty interrupt is generated. When the RSPI is initialized, in order to disable any transmit buffer empty interrupt, 0 should be written to the SPTIE bit simultaneously with the writing of 0 to the SPE bit. To disable any transmit buffer empty interrupt after a mode fault error is detected, use an error handling routine to write 0 to the SPTIE bit.

Corrections

The SPSR.SPRF, SPSR.OVRF, SPSR.MODF, and SPSR.PERF flags are not initialized, nor is the value of the RSPI sequence status register (SPSSR) initialized. For this reason, even after the RSPI is initialized, data from the receive buffer can be read in order to check the status of error occurrence during an RSPI transfer.
The transmit buffer is initialized to an empty state (the SPTEF flag is 1). Therefore, if the SPCR.SPTIE bit is set to 1 after RSPI initialization, a transmit buffer empty interrupt is generated. When the RSPI is initialized, in order to disable any transmit buffer empty interrupt, 0 should be written to the SPTIE bit simultaneously with the writing of 0 to the SPE bit.

•Page 865, “Master Mode Operation” is corrected as follows:

Before correction

(1) Starting a Serial Transfer

The RSPI updates the data in the transmit buffer (SPTX) when data is written to the RSPI data register (SPDR) with the RSPI transmit buffer being empty (data for the next transfer is not set). When the shift register is empty after the number of frames set in the SPDCR.SPFC[1:0] bits are written to the SPDR, the RSPI copies data from the transmit buffer to the shift register and starts serial transfer. Upon copying transmit data to the shift register, the RSPI changes the status of the shift register to “full”, and upon termination of serial transfer, it changes the status of the shift register to “empty”. The status of the shift register cannot be referenced.

Corrections

(1) Starting a Serial Transfer

The RSPI updates the data in the transmit buffer (SPTX) when data is written to the RSPI data register (SPDR) with the RSPI transmit buffer being empty (the SPTEF flag is 1 and data for the next transfer is not set). When the shift register is empty after the number of frames set in the SPDCR.SPFC[1:0] bits are written to the SPDR, the RSPI copies data from the transmit buffer to the shift register and starts serial transfer. Upon copying transmit data to the shift register, the RSPI changes the status of the shift register to “full”, and upon termination of serial transfer, it changes the status of the shift register to “empty”. The status of the shift register cannot be referenced.

Before correction

(2) Terminating a Serial Transfer

Irrespective of the SPCMDm.CPHA bit, the RSPI terminates the serial transfer after transmitting an RSPCKA edge corresponding to the final sampling timing. If free space is available in the receive buffer (SPRX), upon termination of serial transfer, the RSPI copies data from the shift register to the receive buffer of the SPDR register.

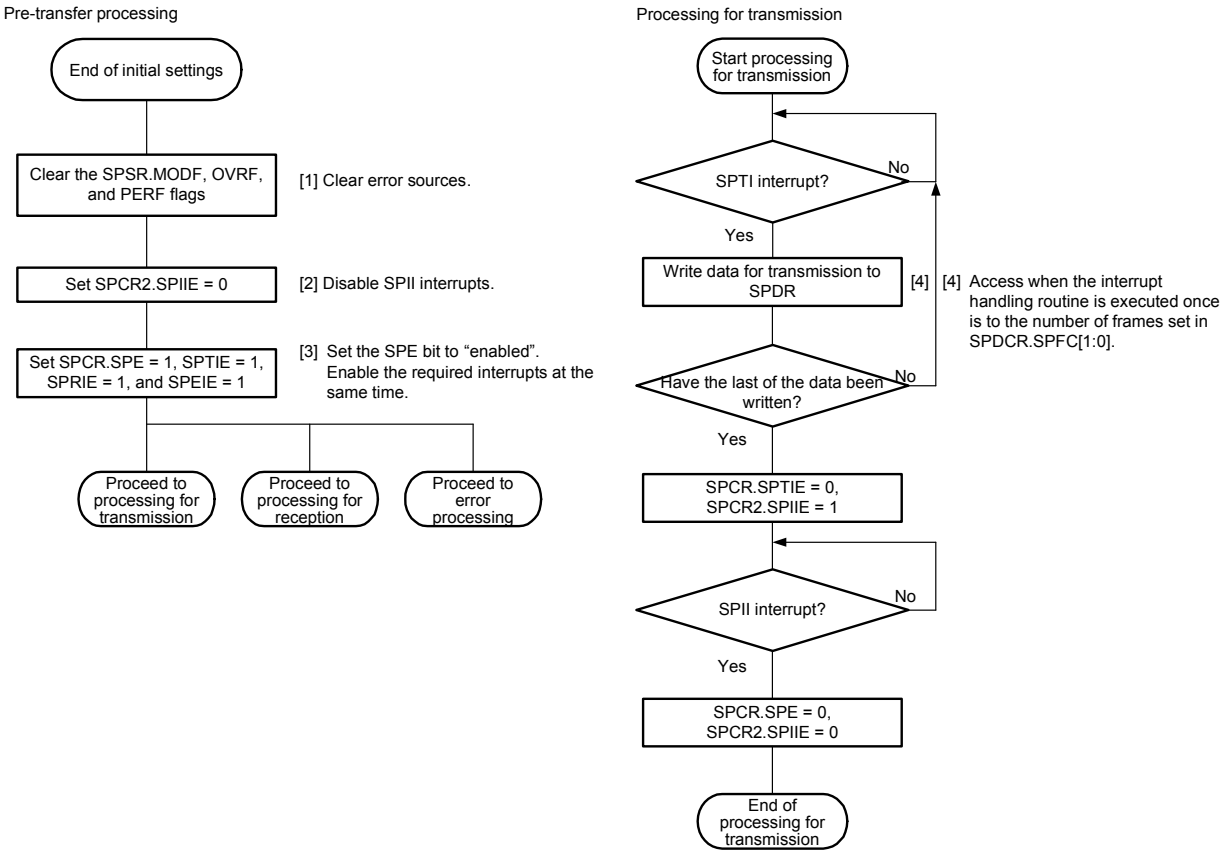
Corrections

(2) Terminating a Serial Transfer

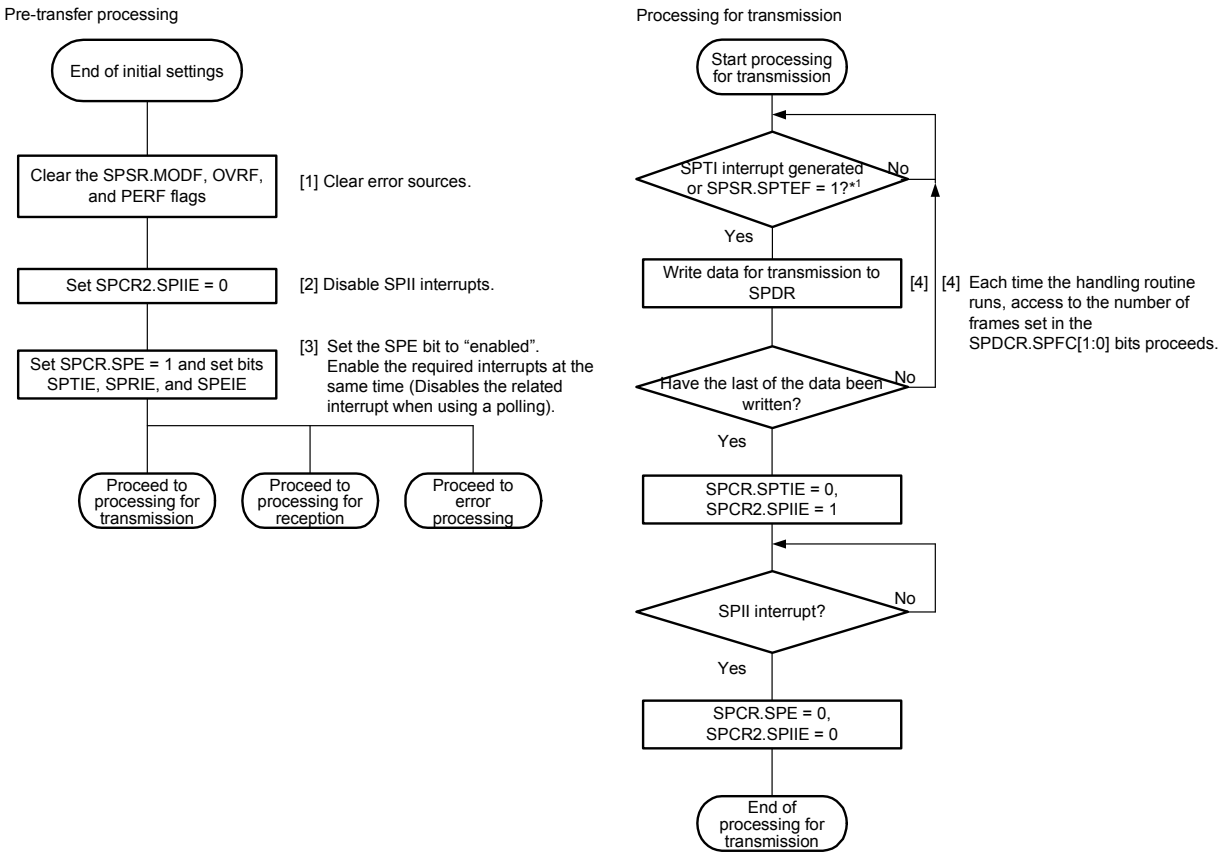
Irrespective of the SPCMDm.CPHA bit, the RSPI terminates the serial transfer after transmitting an RSPCKA edge corresponding to the final sampling timing. If free space is available in the receive buffer (SPRX) (the SPRF flag is 0), upon termination of serial transfer, the RSPI copies data from the shift register to the receive buffer of the SPDR register.

•Page 872, Figure 27.36 Flowchart in Master Mode (Transmission) is corrected as follows:

Before correction



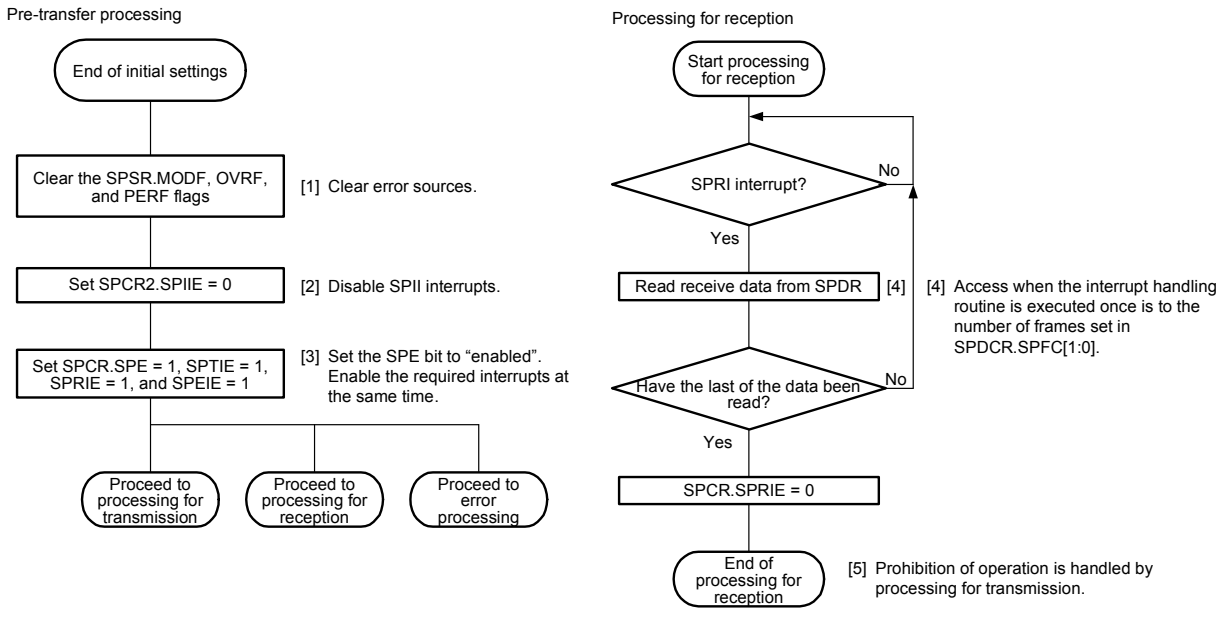
Corrections



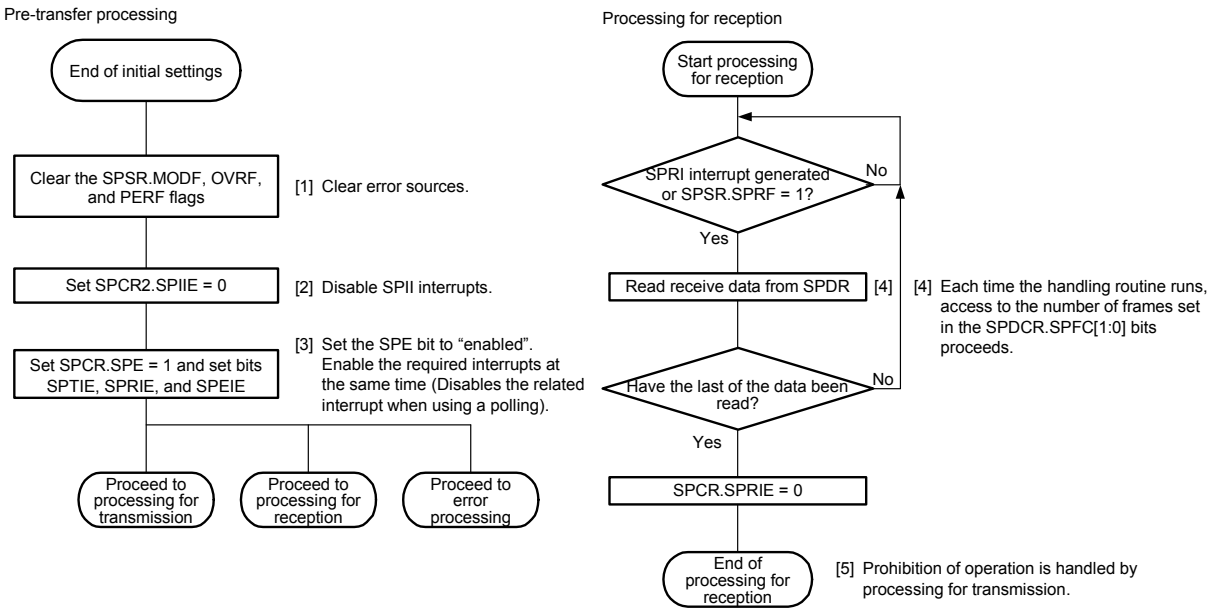
Note 1. When using a polling for the SPTEF flag, proceed to the transmit data writing process after reading the SPTEF flag as 1.

•Page 873, Figure 27.37 Flowchart in Master Mode (Reception) is corrected as follows:

Before correction

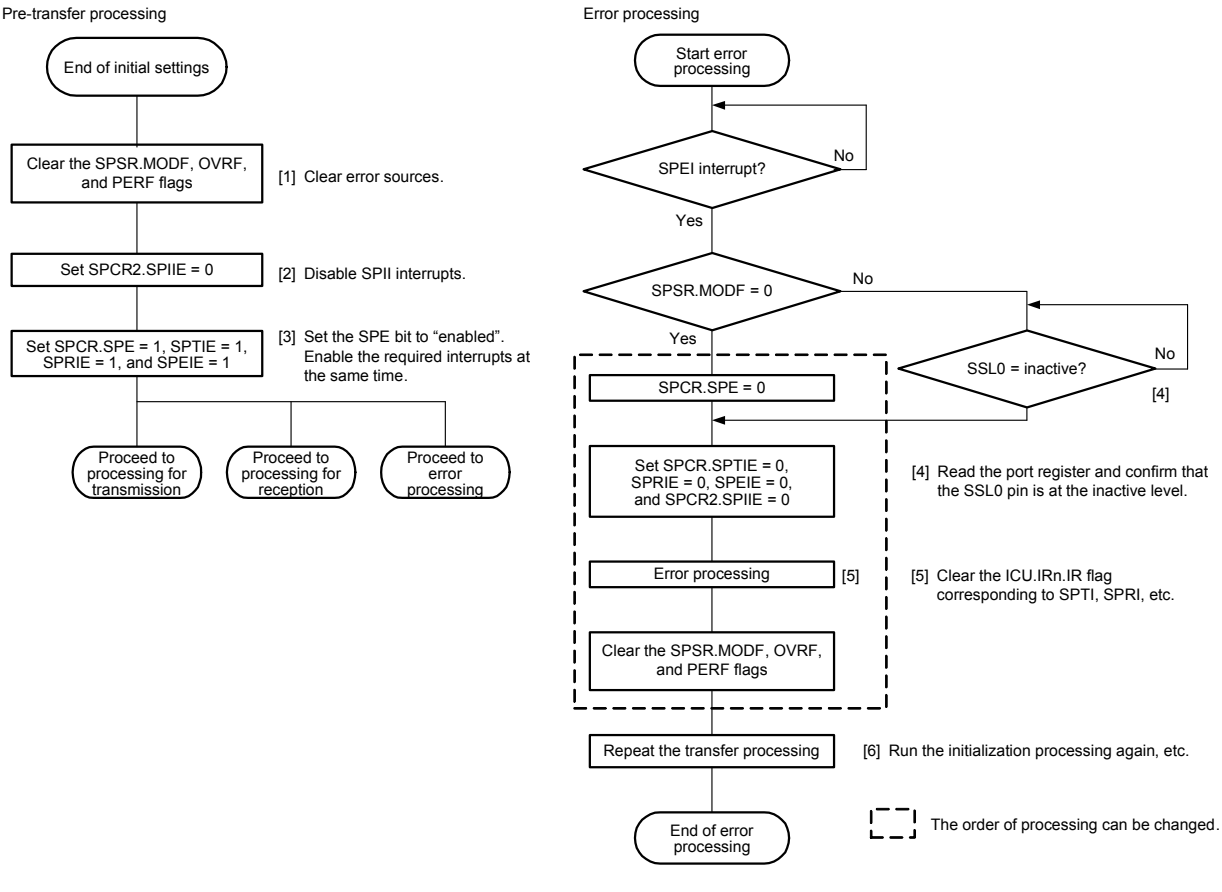


Corrections

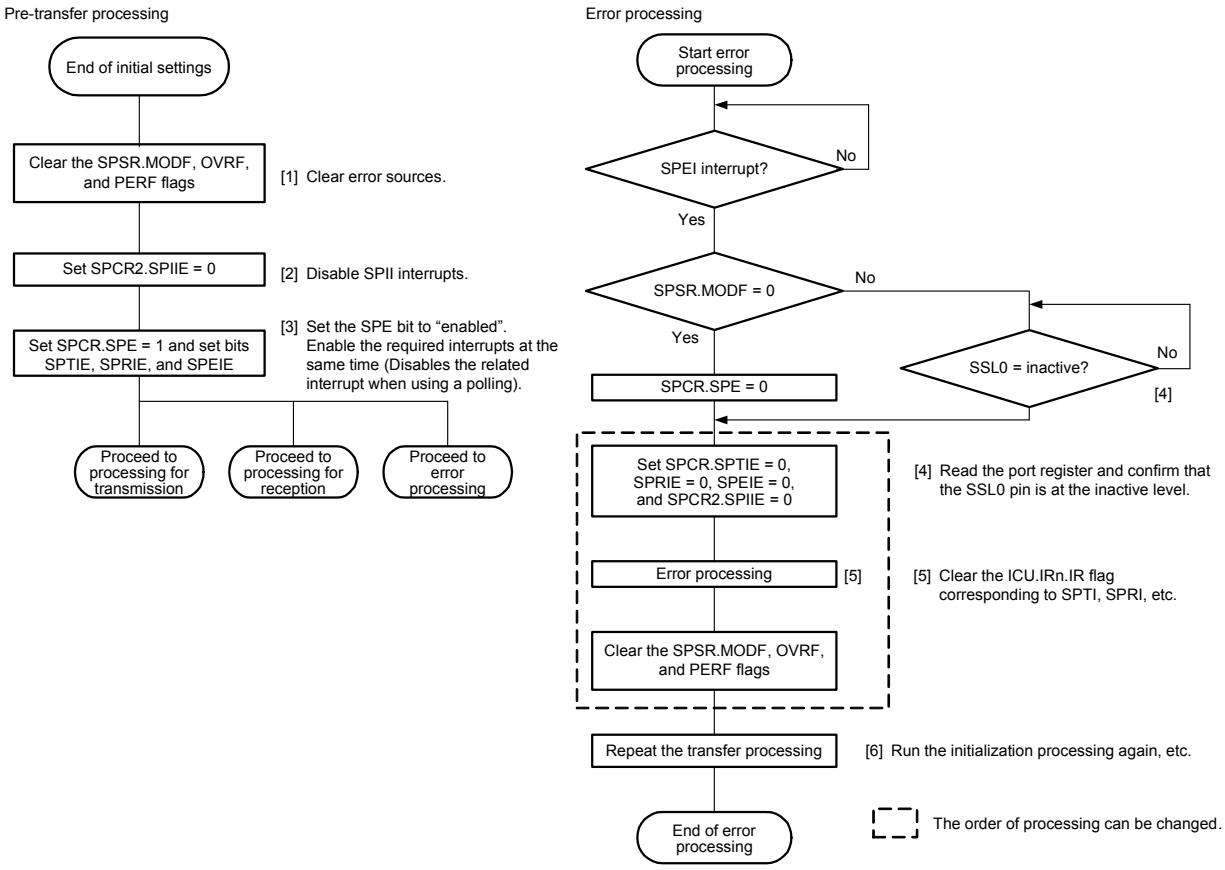


•Page 874, Figure 27.38 Flowchart for Master Mode (Error Processing) is corrected as follows:

Before correction



Corrections



•Page 875, "Slave Mode Operation" is corrected as follows:

Before correction

(2) Terminating a Serial Transfer

Irrespective of the SPCMD0.CPHA bit, the RSPI terminates the serial transfer after detecting an RSPCKA edge corresponding to the final sampling timing. When free space is available in the receive buffer, upon termination of serial transfer the RSPI copies received data from the shift register to the receive buffer of the SPDR register. Upon termination of a serial transfer the RSPI changes the status of the shift register to "empty", regardless of the receive buffer state. A mode fault error occurs if the RSPI detects an SSLA0 input signal negation from the beginning of serial transfer to the end of serial transfer (refer to section 27.3.8, Error Detection).

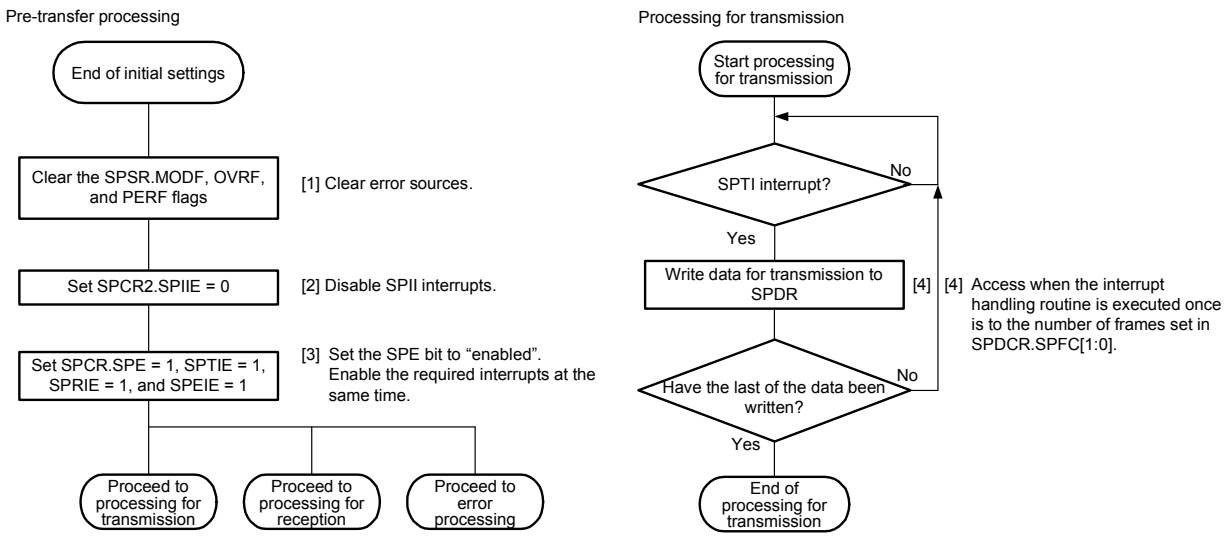
Corrections

(2) Terminating a Serial Transfer

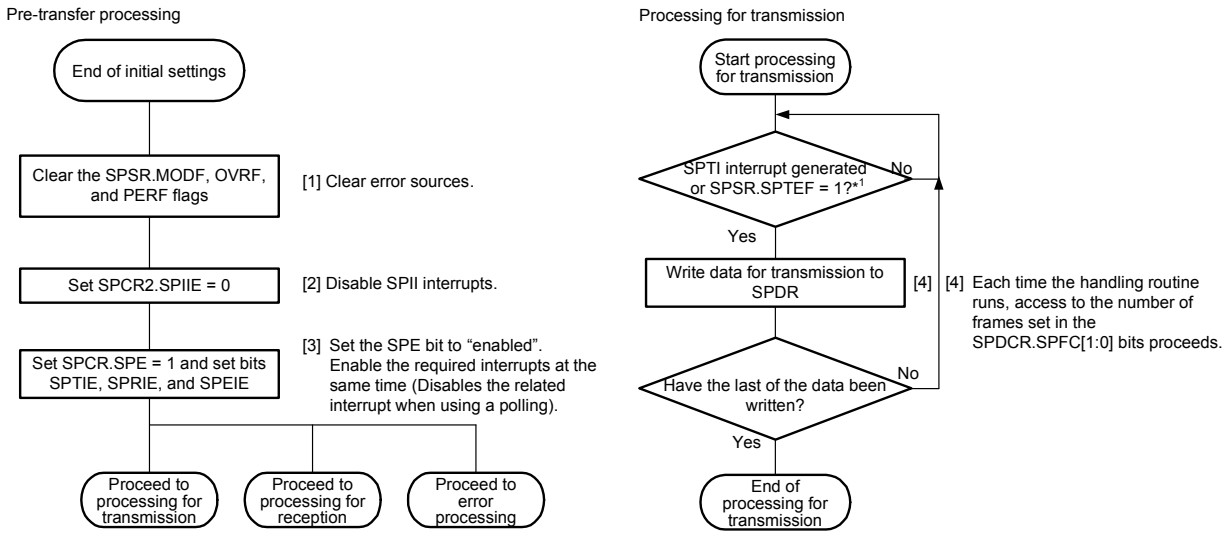
Irrespective of the SPCMD0.CPHA bit, the RSPI terminates the serial transfer after detecting an RSPCKA edge corresponding to the final sampling timing. When free space is available in the receive buffer (the SPRF flag is 0), upon termination of serial transfer the RSPI copies received data from the shift register to the receive buffer of the SPDR register. Upon termination of a serial transfer the RSPI changes the status of the shift register to "empty", regardless of the receive buffer state. A mode fault error occurs if the RSPI detects an SSLA0 input signal negation from the beginning of serial transfer to the end of serial transfer (refer to section 27.3.8, Error Detection).

•Page 877, Figure 27.40 Flowchart in Slave Mode (Transmission) is corrected as follows:

Before correction



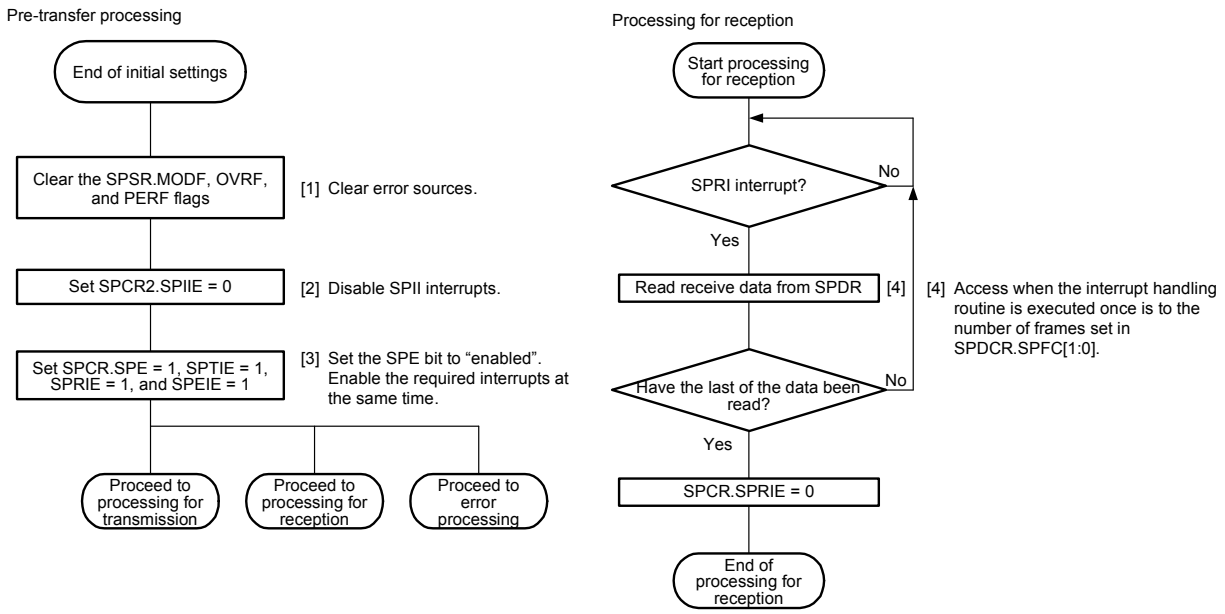
Corrections



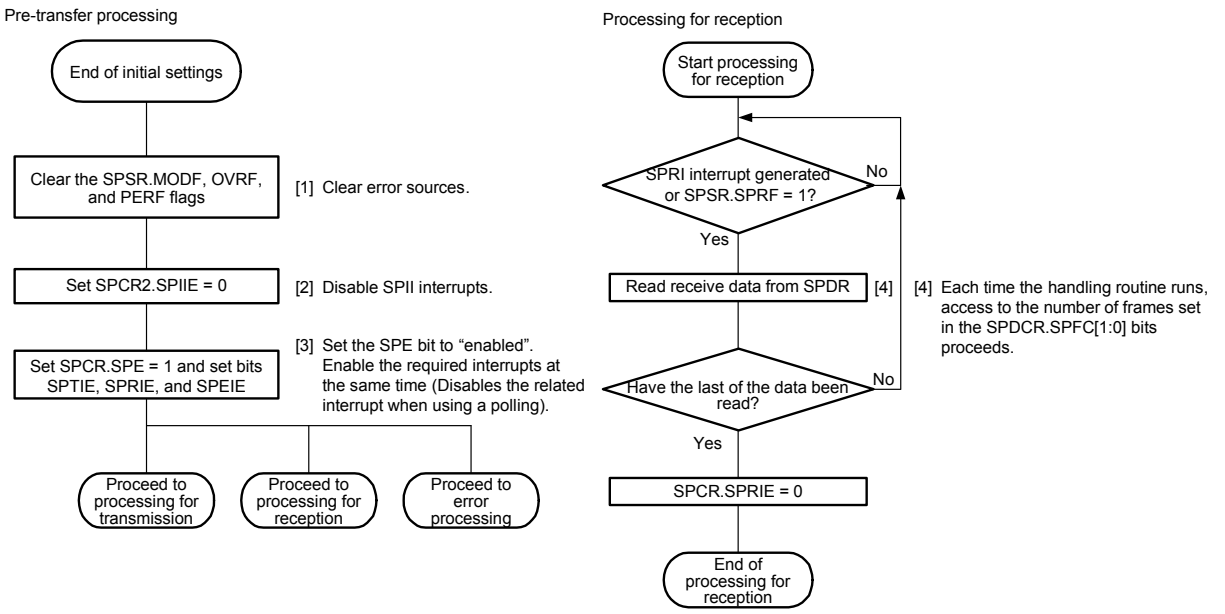
Note 1. When using a polling for the SPTEF flag, proceed to the transmit data writing process after reading the SPTEF flag as 1.

•Page 877, Figure 27.41 Flowchart in Slave Mode (Reception) is corrected as follows:

Before correction

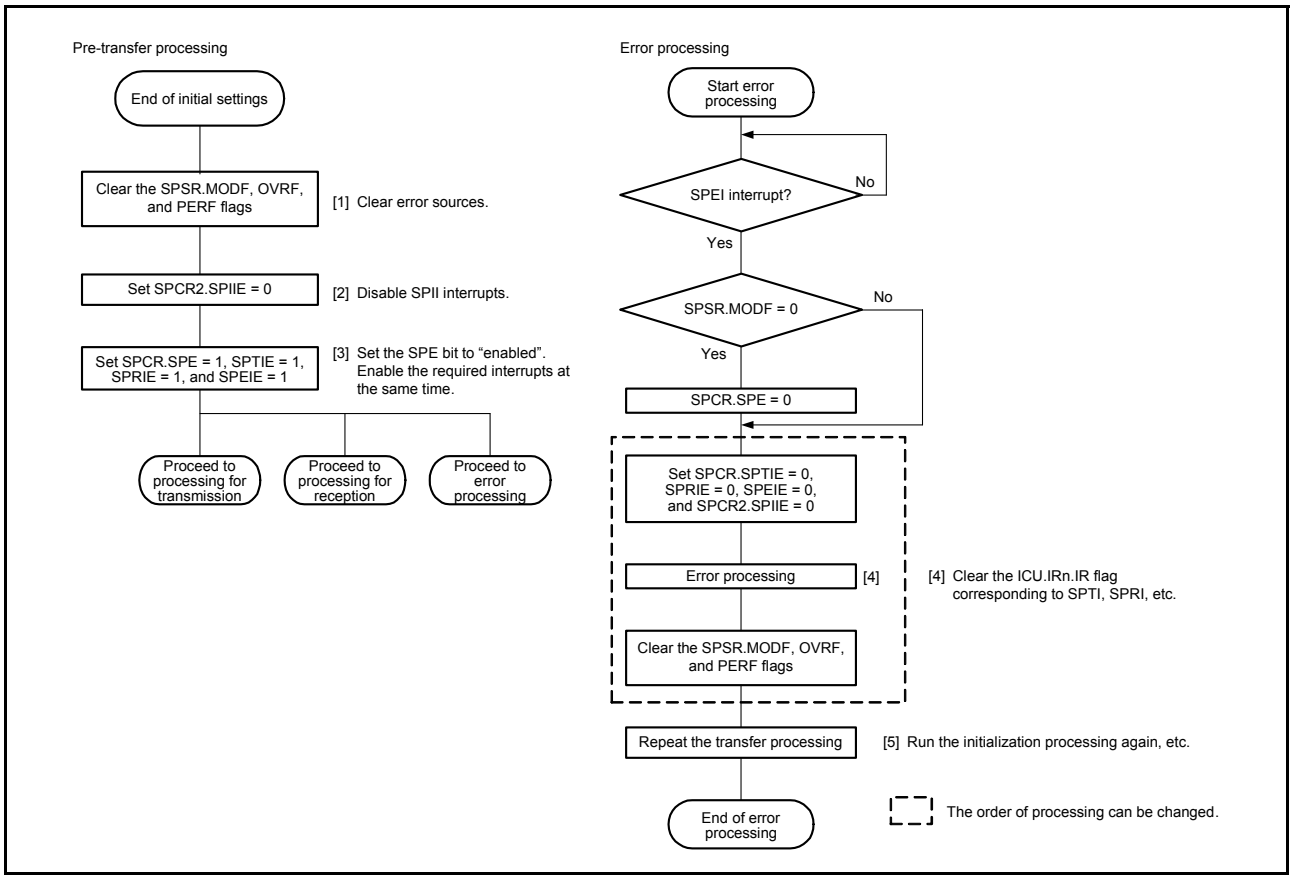


Corrections

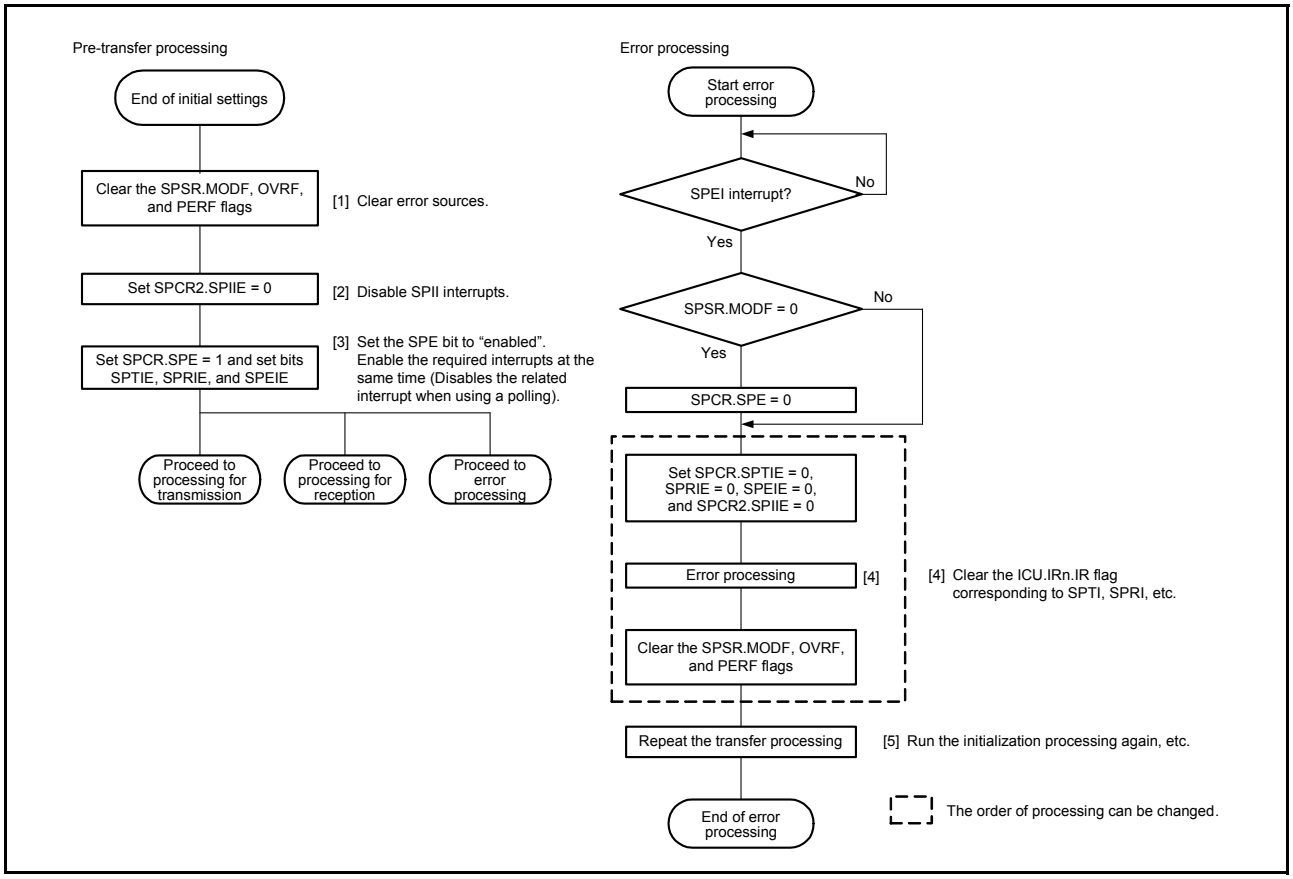


•Page 878, Figure 27.42 Flowchart for Slave Mode (Error Processing) is corrected as follows:

Before correction



Corrections



•Page 879, “Master Mode Operation” is corrected as follows:

Before correction

(1) Starting a Serial Transfer

The RSPI updates the data in the transmit buffer (SPTX) of SPDR when data is written to the SPDR register with the transmit buffer being empty (data for the next transfer is not set).

Corrections

(1) Starting a Serial Transfer

The RSPI updates the data in the transmit buffer (SPTX) of SPDR when data is written to the SPDR register with the transmit buffer being empty (the SPTEF flag is 1 and data for the next transfer is not set).

Before correction

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after transmitting an RSPCKA edge corresponding to the sampling timing. If free space is available in the receive buffer, upon termination of serial transfer, the RSPI copies data from the shift register to the receive buffer of the RSPI data register (SPDR).

Corrections

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after transmitting an RSPCKA edge corresponding to the sampling timing. If free space is available in the receive buffer (SPRX) (the SPRF flag is 0), upon termination of serial transfer, the RSPI copies data from the shift register to the receive buffer of the RSPI data register (SPDR).

•Page 883, “Slave Mode Operation” is corrected as follows:

Before correction

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after detecting an RSPCKA edge corresponding to the final sampling timing. When free space is available in the receive buffer, upon termination of serial transfer the RSPI copies received data from the shift register to the receive buffer of the SPDR register.

Corrections

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after detecting an RSPCKA edge corresponding to the final sampling timing. When free space is available in the receive buffer (the SPRF flag is 0), upon termination of serial transfer the RSPI copies received data from the shift register to the receive buffer of the SPDR register.

•Page 887, Table 27.13 Interrupt Sources of RSPI is corrected as follows:

Before correction

Interrupt Source	Symbol	Interrupt Condition	DTC Activation
Receive buffer full	SPRI	The receive buffer becomes full while the SPCR.SPRIE bit is 1.	Possible
Transmit buffer empty	SPTI	The transmit buffer becomes empty while the SPCR.SPTIE bit is 1.	Possible
RSPI errors (mode fault, overrun and parity error)	SPEI	The SPSR.MODF, OVRF, or PERF flag is set to 1 while the SPCR.SPEIE bit is 1.	Impossible
RSPI idle	SPII	The SPSR.IDLNF flag is set to 0 while the SPCR2.SPIIE bit is 1.	Impossible

Corrections

Interrupt Source	Symbol	Interrupt Condition	DTC Activation
Receive buffer full	SPRI	The receive buffer becomes full (the SPRF flag becomes 1) while the SPCR.SPRIE bit is 1.	Possible
Transmit buffer empty	SPTI	The transmit buffer becomes empty (the SPTEF flag becomes 1) while the SPCR.SPTIE bit is 1.	Possible
RSPI errors (mode fault, overrun and parity error)	SPEI	The SPSR.MODF, OVRF, or PERF flag is set to 1 while the SPCR.SPEIE bit is 1.	Impossible
RSPI idle	SPII	The SPSR.IDLNF flag is set to 0 while the SPCR2.SPIIE bit is 1.	Impossible

•Page 888, “Notes on the SPRF and SPTEF flags” is added as follows:

27.4.4 Notes on the SPRF and SPTEF flags

When polling the SPSR.SPRF flag and/or SPSR.SPTEF flag, set the SPCR.SPRIE bit and/or SPCR.SPTIE bit to 0.

•Page 907, “A/D Channel Select Register A0 (ADANSA0)” is corrected as follows:

Before correction

ANSA0[7:0] Bits (A/D Conversion Channel Select)

Corrections

ANSA0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

•Page 908, “A/D Channel Select Register A1 (ADANSA1)” is corrected as follows:

Before correction

ANSA1[1:0] Bits (A/D Conversion Channel Select)

Corrections

ANSA1n Bit (n = 00, 01) (A/D Conversion Channel Select)

•Page 909, “A/D Channel Select Register B0 (ADANSB0)” is corrected as follows:

Before correction

ANSB0[7:0] Bits (A/D Conversion Channel Select)

Corrections

ANSB0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

•Page 910, “A/D Channel Select Register B1 (ADANSB1)” is corrected as follows:

Before correction

ANSB1[1:0] Bits (A/D Conversion Channel Select)

Corrections

ANSB1n Bit (n = 00, 01) (A/D Conversion Channel Select)

•Page 911, “A/D-Converted Value Addition/Average Function Select Register 0 (ADADS0)” is corrected as follows:

Before correction

ADS0[7:0] Bits (A/D-Converted Value Addition/Average Channel Select)

Corrections

ADS0n Bit (n = 00 to 07) (A/D-Converted Value Addition/Average Channel Select)

•Page 912, “A/D-Converted Value Addition/Average Function Select Register 1 (ADADS1)” is corrected as follows:

Before correction

ADS1[1:0] Bits (A/D-Converted Value Addition/Average Channel Select)

Corrections

ADS1n Bit (n = 00, 01) (A/D-Converted Value Addition/Average Channel Select)

•Page 915, “A/D Control Extended Register (ADCER)” is corrected as follows:

Before correction

The DIAGLD bit should be set while the ADST bit is 0.

Corrections

The DIAGLD bit should be set while the ADCSR.ADST bit is 0.

Before correction

The DIAGLD bit should be set while the ADST bit is 0.

Corrections

The DIAGLD bit should be set while the ADCSR.ADST bit is 0.

Before correction

The ADRFMT bit should be set while the ADST bit is 0.

Corrections

The ADRFMT bit should be set while the ADCSR.ADST bit is 0.

•Page 954 and 955, “Voltage Range of Analog Power Supply Pins” is corrected as follows:

Before correction

- Analog input voltage range
  Voltage applied to analog input pins ANn: VREFL0 ≤ VAN ≤ VREFH0
  Reference voltage range applied to the VREFH0 pin: VREFH0 ≤ AVCC0
  Voltage applied to analog input pins ANn (n = 000 to 007): AVSS0 ≤ VAN ≤ AVCC0
  Voltage applied to analog input pins ANn (n = 016, 017): VSS ≤ VAN ≤ VCC and AVSS0 ≤ VAN ≤ AVCC0

Corrections

- Analog input voltage range
  Voltage applied to analog input pins ANn: AVSS0 ≤ VAN ≤ AVCC0
  Reference voltage range applied to pins VREFH0 and VREFL0: VREFH0 ≤ AVCC0, VREFL0 = AVSS0
  Conversion will not succeed if the voltage applied to analog input pins ANn is greater than VREFH0 (see Figure 29.28).

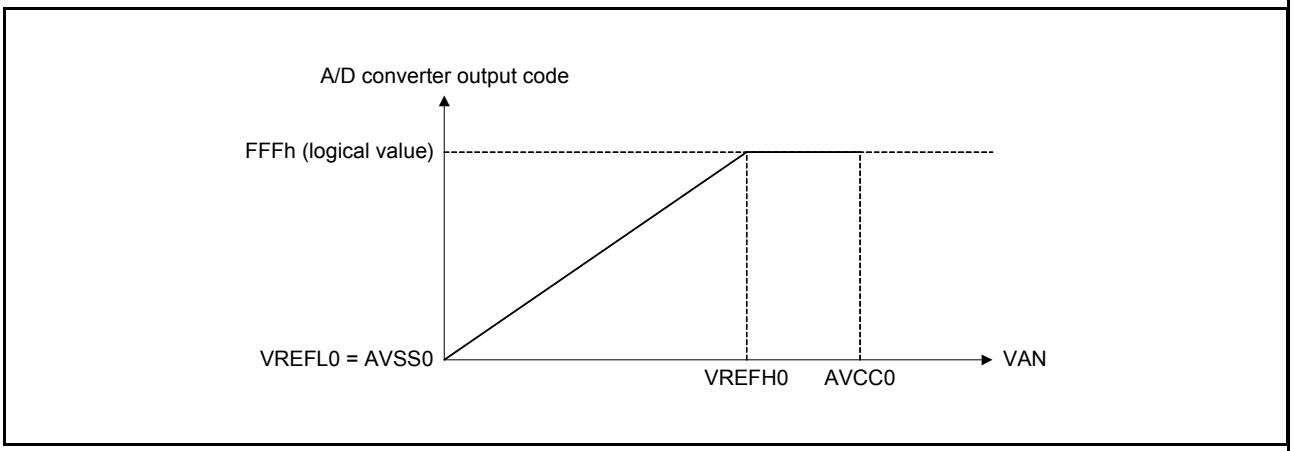


Figure 29.28 Relationship Between Voltage Applied to Analog Input Pins and Output Code

•Page 961, “Setting the D/A Converter” is added as follows:

30.4.4 Setting the D/A Converter

Set the D/A converter for generating comparator C reference voltage and wait for the D/A converter output settling time (t_{DCONV}) before enabling the comparator. Similarly, before making any changes to the settings of the D/A converter stop the comparator temporarily, and after the changes are made, wait for the D/A converter output settling time before enabling the comparator.

•Page 965, “Comparator Control Register (CMPCTL)” is corrected as follows:

Before correction

CEG[1:0] Bits (Comparator Edge Select)

The valid edge is set for the signal after the comparator polarity is selected by the CINV bit and the filter is selected by CDFS[1:0] bits.

Corrections

CEG[1:0] Bits (Comparator Edge Select)

These bits select which edge of comparator output signal is used to generate an interrupt request.

The valid edge is set for the signal after the comparator polarity is selected by the CINV bit and the filter is selected by CDFS[1:0] bits.

•Page 967, “Comparator Reference Voltage Select Register (CMPSEL1)” is corrected as follows:

Before correction

Reference Voltage

Corrections

Reference Input Voltage

Before correction

Note 1. When the internal reference voltage is used, set the D/A converter used for generating the internal reference voltage before enabling comparator operation (CMPCTL.HCMPON bit = 1). For details on setting the internal reference voltage, see section 30, D/A Converter (DA) for Generating Comparator C Reference Voltage.

Corrections

Note 1. When the on-chip D/A converter output voltage is used, set the D/A converter for generating comparator C reference voltage before enabling comparator operation (CMPCTL.HCMPON bit = 1). For details on setting the D/A converter, refer to section 30, D/A Converter for Generating Comparator C Reference Voltage (DA).

•Page 969, “Initial Setting” is corrected as follows:

Before correction

31.3.1 Initial Setting

Figure 31.2 shows an operation example of the comparator. The COMPn (n = 0 to 2) output becomes 1 when the analog input voltage is higher than the reference voltage, and the COMPn output becomes 0 when the analog input voltage is lower than the reference voltage. When the comparator output changes, an interrupt request is output.

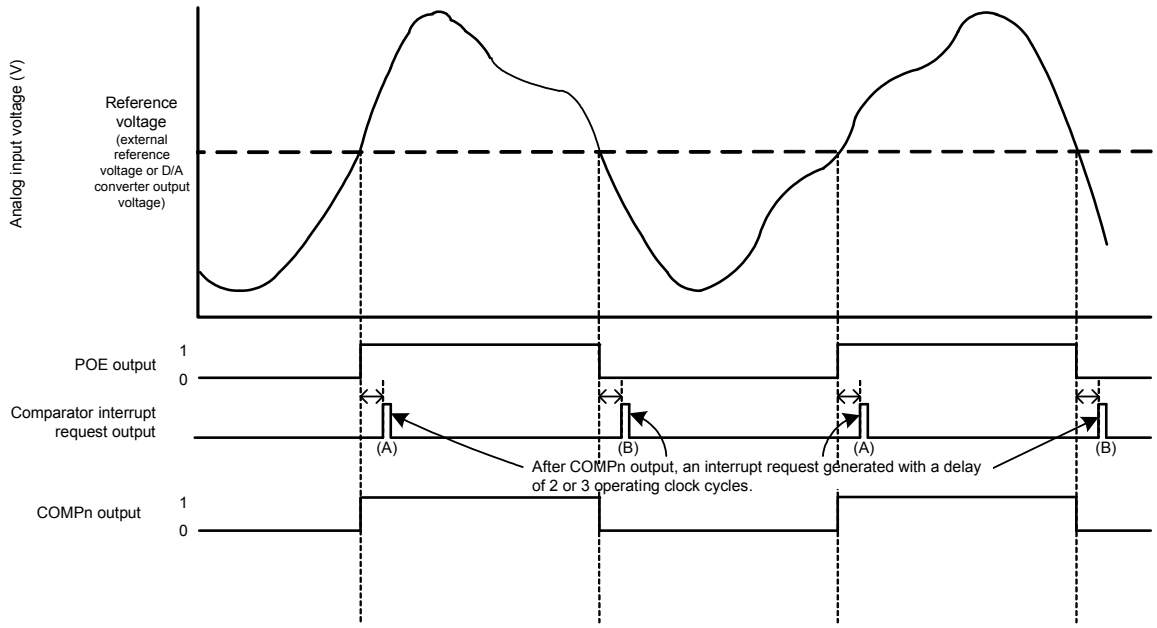
Corrections

31.3.1 Comparator Operation Example

Figure 31.2 shows an operation example of the comparator. The COMPn (n = 0 to 2) output becomes high when the analog input voltage is higher than the reference input voltage, and the COMPn output becomes low when the analog input voltage is lower than the reference input voltage (when the CMPCTL.CINV bit is 0). When the comparator output changes, an interrupt request is output.

•Page 969, Figure 31.2 Comparator Operation Example is corrected as follows:

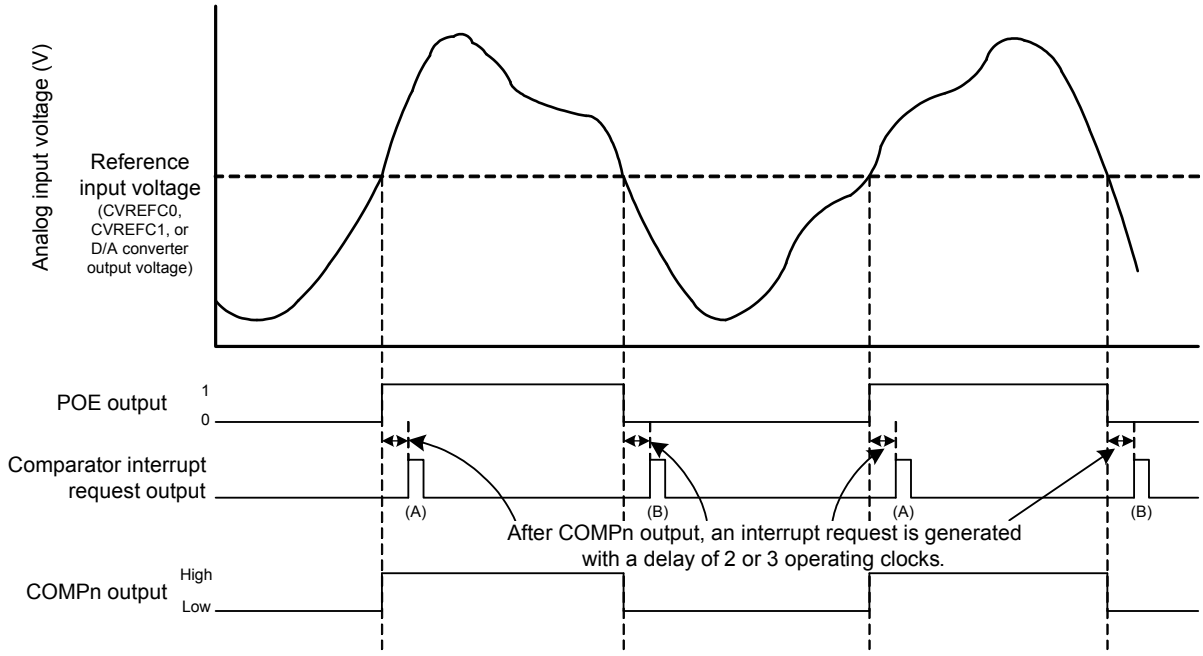
Before correction



n = 0 to 2

Note: The above diagram applies when the CMPIOC.CPOE bit = 1 (pin output enabled), the CMPCTL.CDFS[1:0] bits = 00b (filter not used), and the CMPCTL.CEG[1:0] bits = 11b (both edges selected).

Corrections

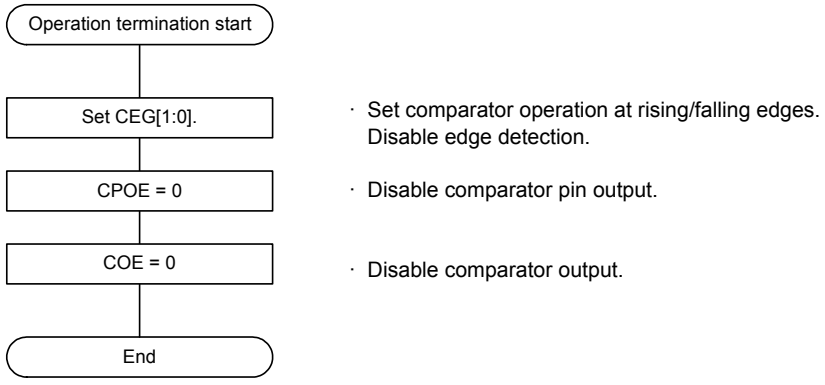


n = 0 to 2

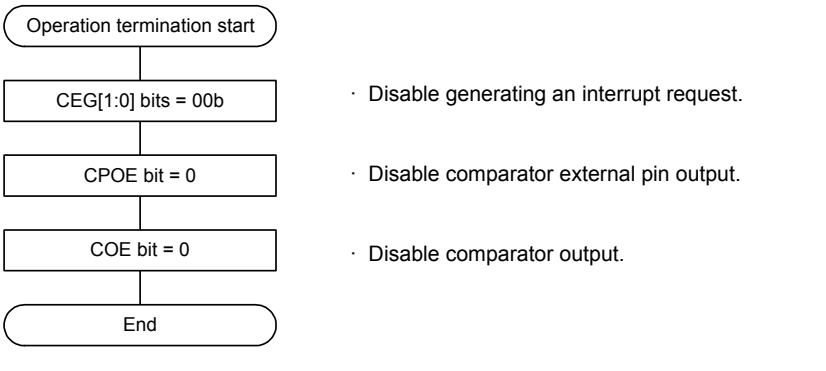
Note: The above diagram applies when the CMPIOC.CPOE bit is 1 (pin output enabled), the CMPCTL.CDFS[1:0] bits are 00b (filter not used), the CMPCTL.CEG[1:0] bits are 11b (both edges selected), and the CMPCTL.CINV bit is 0 (comparator output not inverted).

•Page 972, Figure 31.6 Comparator Operation Termination Flowchart is corrected as follows:

Before correction



Corrections



•Page 973, “Comparator C Operation in Software Standby Mode” is corrected as follows:

Before correction

31.4.3 Comparator C Operation in Software Standby Mode

When software standby mode is entered while comparator C is operating, comparator C operation is retained and the amount of the analog power supply current is the same as when comparator C is being used. If the analog power supply current needs to be reduced in software standby mode, set the CMPCTL.HCMPON bit to 0 to stop comparator C.

Corrections

31.4.3 Comparator C Operation in Software Standby Mode

When software standby mode is entered while comparator C is operating, analog circuits in the comparator C is not stopped and the analog power supply current is the same as that when comparator C is being used. If the analog power supply current needs to be reduced in software standby mode, set the CMPCTL.HCMPON bit to 0 to stop comparator C.

•Page 973, “Setting the D/A Converter for Generating Comparator C Reference Voltage” is added as follows:

31.4.4 Setting the D/A Converter for Generating Comparator C Reference Voltage

Set the D/A converter for generating comparator C reference voltage and wait for the D/A converter output settling time before enabling the comparator. Similarly, before making any changes to the settings of the D/A converter stop the comparator temporarily, and after the changes are made, wait for the D/A converter output settling time before enabling the comparator.

•Page 980, Table 33.1 Specifications of RAM is corrected as follows:

Before correction

Item	Description
RAM capacity	10 Kbytes (RAM0: 10 Kbytes)
RAM address	RAM0: 0000 0000h to 0000 27FFh
Access	• Single-cycle access is possible for both reading and writing. • On-chip RAM can be enabled or disabled.*1
Low power consumption function	The module stop state is selectable for RAM0.

Corrections

Item	Description
RAM capacity	12 Kbytes (RAM0: 12 Kbytes)
RAM address	RAM0: 0000 0000h to 0000 27FFh, 0000 4000h to 0000 4A7Fh
Access	• Single-cycle access is possible for both reading and writing. • On-chip RAM can be enabled or disabled.*1
Low power consumption function	The module stop state is selectable for RAM0.

•Page 1024, “Block Information Inquiry” is corrected as follows:

Before correction

Command	26h		
Response	36h	Size	DDh
	Start address of the user area		
	Block size of one block for the user area		
	Number of blocks of the user area		
	SUM		

Corrections

Command	26h		
Response	36h	Size	DDh
	Start address of the user area		
	Block size of one block for the user area		
	Number of blocks of the user area		
	Dummy data		
	Dummy data		
	Dummy data		
	SUM		

•Page 1037, “Serial Programmer Operation in Boot Mode (SCI)” is corrected as follows:

Before correction

Note 1. If the necessary information has been already received, step 2 can be skipped.
Note 2. Any step from 6 to 10 can be skipped, and their order can be changed.
Note 3. When a timeout occurs or invalid response data is received, stop the operation and perform step 11 (reset the MCU).

Corrections

Note 1. If the necessary information has been already received, step 2 can be skipped.
Note 2. Any step from 6 to 9 can be skipped, and their order can be changed.
Note 3. When a timeout occurs or invalid response data is received, stop the operation and perform step 10 (reset the MCU).

Before correction

Refer to section 34.9.5, Inquiry Commands, section 34.9.6, Setting Commands, section 34.9.7, ID Code Authentication Command, section 34.9.8, Program/Erase Commands, and section 34.9.9, Read-Check Commands for details on the commands used in the above steps 2 to 10.

Corrections

Refer to section 34.9.5, Inquiry Commands, section 34.9.6, Setting Commands, section 34.9.7, ID Code Authentication Command, section 34.9.8, Program/Erase Commands, and section 34.9.9, Read-Check Commands for details on the commands used in the above steps 2 to 9.

•Page 1097, Figure 35.54 Connecting Capacitors (64 Pins) is corrected as follows:

Before correction

PLQP0064KB-A

Corrections

PLQP0064KB-C

•Page 1098, Figure 35.55 Connecting Capacitors (52 Pins) is corrected as follows:

Before correction

PLQP0052JA-A

Corrections

PLQP0052JA-B

•Page 1099, Figure 35.56 Connecting Capacitors (48 Pins) is corrected as follows:

Before correction

PLQP0048KB-A

Corrections

PLQP0048KB-B

•Page 1101, Appendix 2. Package Dimensions is corrected.