(1/2)

# **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-RL*-A0139A/E	Rev.	1.00
Title	Correction for Incorrect Description Notice Descriptions of errors in the RL78/F23, F24 L Manual: Hardware Rev.1.00	Jser's	Information Category	Technical Notification		
		Lot No.				
Applicable Product	RL78/F23, F24 Group	All lots	Reference Document	RL78/F23, F24 User's Hardware Rev.1.00 (R01UH0944EJ0100)	Manual:	

This document describes misstatements found in RL78/F23, F24 User's Manual: Hardware Rev.1.00 (R01UH0944JJ0100).

These corrections will be made for the next revision of the User's Manual: Hardware Rev.1.10 (R01UH0944JJ0110).

### Corrections:

			(1/3)
No	Corrections	R01UH0944EJ0100	Pages in this document
1	Figure 3-1 and Figure 3-2: Correct note in the memory map.	P.69, 70	P.4
2	Figure 4-96: Correct the typo in the reserved word definition for PMS0 bit.	P.263	P.6
3	Table 4-26: Correct typo of the POMm, PIMm, and PITHLm registers corresponding pin.	P.267 to 274	P.7
4	Add new section "4.6.4 Cautions when setting peripheral I/O redirection registers".	_	P.8
5	Figure 6-66: Correct the starting point of the arrow when restarting the operation.	P.437	P.9
6	Figure 9-8 (2/2): Change the explanation of RWAIT bit.	P.671	P.11
7	Figure 9-23: Change the description of the "Caution" below the figure.	P.684	P.12
8	Figure 9-24: Change the description of the "Caution 1" below the figure.	P.685	P.13
9	Figure 12-16: Add the Note 1 to the ADNDIS[4] bit and Note 2 to the ADNDIS[3:0] bits, and listed the notes separately.	P.729	P.14
10	Figure 12-22: Correct the typo in the setting value of ADPAGE[3:0] bits when accessing ADSSTRL and ADSSTRO registers.	P.737	P.16
11	Figure 14-3: Correct a typo in Note 5 on the CSTEN bit.	P.784	P.17
12	Section 15.1.2: Add "Reception end interrupt" to [Interrupt function].	P.798	P.17
13	Figure 15-9: Correct the typo in the register symbol in "Caution 2".	P.815	P.18
14	Figure 15-10: Correct the typo in the bit symbol in "Caution".	P.816	P.19
15	Section 15.3.19: Add the description of serial data input/output (SDAmn) to the text. Also add the setting example when using the SDAmn function.	P.829	P.19
16	Figure 15-75: Correct a typo in SOmn output.	P.901	P.20
17	Figure 15-80: Add slave selection signal output pin to the figure.	P.908	P.22
18	Figure 15-82: Add slave selection signal output pin to the figure.	P.910	P.23
19	Figure 15-88: Add the slave selection signal output pin to the figure.	P.917	P.24
20	Figure 15-90: Add the slave selection signal output pin to the figure.	P.919	P.25
21	Figure 15-96: Add the slave selection signal output pin to the figure.	P.928	P.26
22	Figure 15-98: Add the slave selection signal output pin to the figure.	P.930	P.27
23	Figure 15-100: Change the register setting order in the figure in the same way as in the Figure 15-101.	P.934	P.28
24	Figure 15-102: Correct the content in "Remark 1" below the figure.	P.936	P.29



No	Corrections	R01UH0944EJ0100	(2/3 Pages in this document
25	Figure 15-104: Add the SSIp pin in the figure. Also correct the output of the SOp pin.	P.938	P.30
26	Figure 15-106: Add the SSIp pin in the figure. Also correct the output of the SOp pin.	P.940	P.31
27	Figure 15-108: Change the register setting order in the figure in the same way as in the Figure 15-109.	P.944	P.32
28	Figure 15-112: Add the SSIp pin in the figure.	P.947	P.33
29	Figure 15-114: Change the register setting order in the figure in the same way as in the Figure 15-115.	P.951	P.34
30	Figure 15-118: Add the SSIp pin in the figure. Also correct the output of the SOp pin.	P.955	P.35
31	Figure 15-120: Add the SSIp pin in the figure. Also correct the output of the SOp pin.	P.957	P.36
32	Section 15.7: Add "Reception end interrupt" to [Interrupt function].	P.962	P.37
33	Figure 16-33 (1): Correct reference figure number.	P.1101	P.37
34	Section 17.2.1 (16): Add "Caution" regarding LIN reset mode transitions.	P.1129	P.37
35	Section 17.2.2 (3): Correct typos in the "Caution" below the table.	P.1143	P.38
36	Section 17.2.2 (15): Add "Caution" regarding LIN reset mode transitions.	P.1156	P.38
37	Section 17.2.3 (3): Correct typos in the "Caution" below the table.	P.1172	P.39
38	Section 17.3.1: Add the "Caution" about transitioning to LIN reset mode when timeout error detection is enabled.	P.1202	P.40
39	Section 18.3.5: Correct the typo in the DBRP[7:0] bit setting value.	P.1295	P.41
40	Table 19-1: Add description of high-speed DTC to the specification explanation for "Repeat mode" in "Transfer mode".	P.1500	P.42
41	Section 19.3.4: Add Note regarding DTCCR23 register.	P.1529	P.42
42	Section 19.4.2: Correct typo in general-purpose register addresses.	P.1531	P.42
43	Figure 21-2: Correct the typo in the reserved word definition for TMIF12 bit.	P.1561	P.43
44	Figure 21-6: Delete unnecessary references in "Note 4" below the figure.	P.1569	P.43
45	Figure 21-9: Correct typos in the reserved word definition for ISC0, ISC2, and ISC3 bits.	P.1573	P.44
46	Section 21.4.5: Add an example of the state when interrupt request hold instructions are consecutive.	P.1584	P.44
47	Table 24-4: Correct the bit symbol in the Note 2.	P.1620	P.45
48	Section 25.1: Add a description of the RESF register when an internal reset occurs to the Remark.	P.1623	P.45
49	Section 26.1: Correct the fourth bullet point description.	P.1629	P.46
50	Figure 27-37 (a): Add "NOP instruction (1 cycle)" in the figure.	P.1671	P.46
51	Section 27.3.2.4: Correct typos in the internal calculation formula.	P.1676	P.47
52	Section 27.3.2.5: Correct typos in the internal calculation formula.	P.1678	P.48
53	Section 27.3.2.14: Correct a typo in the internal calculation formula.	P.1693	P.49
54	Figure 28-8: Correct a typo in the ERADR register.	P.1707	P.50
55	Section 28.3.5: Correct typo in register symbols of syndrome code in "Operation Explanation 1".	P.1726	P.51
56	Figure 28-43: Correct typo in pin function name.	P.1749	P.52
57	Figure 31-4: Add FLPEN bit to the bit setting combination description.	P.1760	P.53
58	Figure 32-8: Add "HALT mode to prohibited transitions" to flow annotations.	P.1778	P.54
59	Figure 32-13: Correct description of DCLR bit.	P.1784	P.55
60	Section 32.7.2.12: Correct the typo in the description of the DCLR bit of the FSSQ register.	P.1791	P.55
61	Section 32.9.1: Add "HALT mode to prohibited transitions" to bullet point description.	P.1816	P.55
62	Section 35.2: Correct a typo in the operation description of the OR1 instruction.	P.1843	P.56

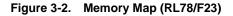


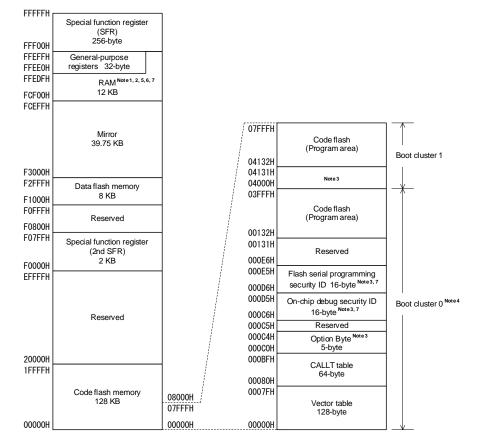
			(3/3)
No	Corrections	R01UH0944EJ0100	Pages in this document
63	Section 36.10: Correct "Note 4" description below the table.	P.1899	P.56
64	Section 36.10: Add "(1) Code flash memory processing time" and "(2)	P.1899	P.57
	Data flash memory processing time" sections.		
65	Section 37.10: Correct "Note 4" description below the table.	P.1953	P.56
66	Section 37.10: Add "(1) Code flash memory processing time" and "(2)	P.1953	P.57
	Data flash memory processing time" sections.		
67	Section 38.10: Correct "Note 4" description below the table.	P.2005	P.56
68	Section 38.10: Add "(1) Code flash memory processing time" and "(2)	P.2005	P.57
	Data flash memory processing time" sections.		
69	Section 39.1: Correct the value of the pin cross section in the figure.	P.2007	P.58

Incorrect: Bold with underline: Correct: Gray hatched

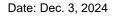


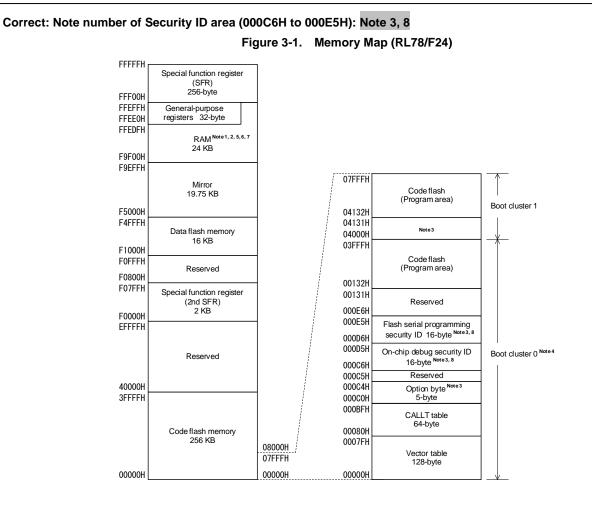
#### No.1: Correct note in the memory map. Page: P.69, 70 Incorrect: Note number of Security ID area (000C6H to 000E5H): Note 3.7 Figure 3-1. Memory Map (RL78/F24) FFFFFH Special function register (SFR) 256-byte FFF00H General-purpose registers 32-byte FFEFFH **FFEE0H** FFEDFH RAM Note 1, 2, 5, 6, 7 24 KB F9F00H F9EFFH 07FFFH Mirror Code flash 19.75 KB (Program area) Boot cluster 1 F5000H 04132H F4FFFH 04131H Note 3 Data flash memory 04000H 16 KB 03FFFH F1000H Code flash FOFFFH Reserved (Program area) F0800H 00132H F07FFH Special function register 00131H (2nd SFR) Reserved 000E6H 2 KB F0000H 000E5H Flash serial programming security ID 16-byte Note 3, 7 EFFFFH 000D6H 000D5H On-chip debug security ID 16-byte Note 3, 7 Boot cluster 0 Note 4 Reserved 000C6H 000C5H Reserved 40000H 000C4H Option byte Note 3 3FFFFH 000C0H 5-bvte 000BFH CALLT table 64-byte 00080H Code flash memory 256 KB 0007FH 08000H Vector table 07FFFH 128-byte 00000H 00000H 00000H



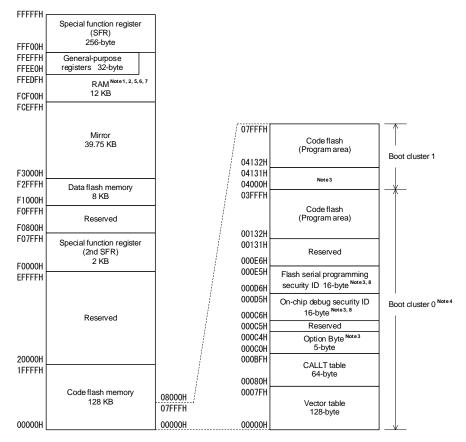














Γ

ect: Figure 4-96. Port Mode Select Register (PMS) Address: F0077H After reset: 00H R/W Symbol 7 6 5 4 3 2 1 0	Figure 4-96. Port Mode Select Register (PMS)         ess:       F0077H       After reset:       00H       R/W         mbol       7       6       5       4       3       2       1       sQ≥         MS       0       0       0       0       0       0       0       PMS0         Figure 4-96. Port Mode Select Register (PMS)         ess: F0077H       After reset:       00H       R/W         mbol       7       6       5       4       3       2       1       0
Address:       F0077H       After reset:       OH       R/W         Symbol       7       6       5       4       3       2       1       state         PMS       0       0       0       0       0       0       0       PMS0         ect:       Figure 4-96. Port Mode Select Register (PMS)         address: F0077H       After reset:       OH       R/W         Symbol       7       6       5       4       3       2       1       0	ess: F0077H After reset: 00H R/W mbol 7 6 5 4 3 2 1 <b>≤0≥</b> MS 0 0 0 0 0 0 0 0 0 PMS0 Figure 4-96. Port Mode Select Register (PMS) ess: F0077H After reset: 00H R/W mbol 7 6 5 4 3 2 1 0
Symbol       7       6       5       4       3       2       1       sile         PMS       0       0       0       0       0       0       0       0       PMS0         ect:       Figure 4-96. Port Mode Select Register (PMS)         address:       F0077H       After reset: 00H       R/W         Symbol       7       6       5       4       3       2       1       0	mbol       7       6       5       4       3       2       1       sQ≥         MS       0       0       0       0       0       0       0       0       PMS0         Figure 4-96. Port Mode Select Register (PMS)         ess: F0077H       After reset: 00H       R/W         mbol       7       6       5       4       3       2       1       0
ect: Figure 4-96. Port Mode Select Register (PMS) Address: F0077H After reset: 00H R/W Symbol 7 6 5 4 3 2 1 0	Figure 4-96.       Port Mode Select Register (PMS)         ess:       F0077H       After reset:       00H       R/W         mbol       7       6       5       4       3       2       1       0
Figure 4-96.       Port Mode Select Register (PMS)         Address:       F0077H       After reset:       00H       R/W         Symbol       7       6       5       4       3       2       1       0	Figure 4-96. Port Mode Select Register (PMS)           ess:         F0077H         After reset:         00H         R/W           mbol         7         6         5         4         3         2         1         0
Figure 4-96.       Port Mode Select Register (PMS)         Address:       F0077H       After reset:       00H       R/W         Symbol       7       6       5       4       3       2       1       0	Figure 4-96. Port Mode Select Register (PMS)           ess:         F0077H         After reset:         00H         R/W           mbol         7         6         5         4         3         2         1         0
Address: F0077H After reset: 00H R/W Symbol 7 6 5 4 3 2 1 0	ess: F0077H After reset: 00H R/W mbol 7 6 5 4 3 2 1 0
Symbol 7 6 5 4 3 2 1 0	mbol 7 6 5 4 3 2 1 0
PMS 0 0 0 0 0 0 0 0 0 PMS0	MS <u>0 0 0 0 0 0 0 0 0 9MS0</u>



# No.3: Correct typo of the POMm, PIMm, and PITHLm registers corresponding pin.

#### Page: P.267 to 274

Incorrect:

#### Table 4-26. Settings of Port Mode Register and Output Latch When Using Alternate Function

	1		-	1	1	1	1	1	1
Pin Name	Alternate Fi	unction	PIORx	POMm	PMCm	PMm	Pm	PIMm	PITHLm
	Function Name	I/O							
P10	LTXD1	Output	PIOR45=0	<u>0</u>	—	0	1	×	×
P11	LRXD1	Input	PIOR45=0	×	-	1	×	<u>0</u>	0/1
	CRXD0	Input	PIOR46=0	×	—	1	×	0	<u>0/1</u>
P13	LTXD0	Output	PIOR44=0	<u>0</u>	_	0	1	×	×
P14	LRXD0	Input	PIOR44=0	×	_	1	×	<u>0</u>	0/1
P41	TI10	Input	PIOR20=0	_	_	1	×	-	<u>0</u>
	TRJIO0	Input	-	_	_	1	×	_	<u>0</u>
P52	(STOPST) Note	Output	-	_	_	0	0	-	<u>0/1</u>
P60	(SCK00)	Output	PIOR40=1	<u>0</u>	—	0	1	-	×
P62	(SO00)	Output	PIOR40=1	<u>0</u>	_	0	1	×	×
	(TXD0)	Output	PIOR40=1	<u>0</u>	_	0	1	×	×
P63	(SSI00)	Input	PIOR40=1	×	_	1	×	<u>0</u>	0/1
P73	(CRXD0)	Input	PIOR46=1	_	0	1	×	0	<u>0/1</u>
P120	(LTXD1)	Output	PIOR45=1,	<u>0</u>	0	0	1	_	×
			PIOR93=1						
P125	(LRXD1)	Input	PIOR45=1,	_	0	1	×	<u>0</u>	0/1
			PIOR93=1						

#### Correct:

#### Table 4-26. Settings of Port Mode Register and Output Latch When Using Alternate Function

Pin Name	Alternate Fu	unction	PIORx	POMm	PMCm	PMm	Pm	PIMm	PITHLm
	Function Name	I/O							
P10	LTXD1	Output	PIOR45=0	0/1	-	0	1	×	×
P11	LRXD1	Input	PIOR45=0	×	—	1	×	0/1	0/1
	CRXD0	Input	PIOR46=0	×	_	1	×	0	0
P13	LTXD0	Output	PIOR44=0	0/1	_	0	1	×	×
P14	LRXD0	Input	PIOR44=0	×	—	1	×	0/1	0/1
P41	TI10	Input	PIOR20=0	_	_	1	×	_	0/1
	TRJIO0	Input	-	_	_	1	×	_	0/1
P52	(STOPST) Note	Output	-	—	—	0	0	—	×
P60	(SCK00)	Output	PIOR40=1	0/1	_	0	1	—	×
P62	(SO00)	Output	PIOR40=1	0/1	_	0	1	×	×
	(TXD0)	Output	PIOR40=1	0/1	—	0	1	×	×
P63	(SSI00)	Input	PIOR40=1	×	—	1	×	0/1	0/1
P73	(CRXD0)	Input	PIOR46=1	_	0	1	×	0	0
P120	(LTXD1)	Output	PIOR45=1,	0/1	0	0	1	—	×
			PIOR93=1						
P125	(LRXD1)	Input	PIOR45=1,	_	0	1	×	0/1	0/1
			PIOR93=1						



# No.4: Add new section "4.6.4 Cautions when setting peripheral I/O redirection registers".

Page: –

# Correct:

#### 4.6.4 Cautions When Setting Peripheral I/O Redirection Register

Some pin functions may require multiple peripheral I/O redirection register (PIORx) settings when assigned to a port. These functions should be configured with care in combination.

#### • TO01, TO02, TO03, TO07 (Timer output pin for timer array unit 0)

Note that PIOR90 bit selects these functions (TO01, TO02, TO03, and TO07) together.

Pin Function	Assigned Port Name (Condition of PIORx Register)			
TO01	P30 (PIOR11 = 0)	P126 (PIOR11 = 1, PIOR90 = 0)	P60 (PIOR11 = 1, PIOR90 = 1)	
TO02	P16 (PIOR12 = 0)	P67 (PIOR12 = 1, PIOR90 = 0)	P61 (PIOR12 = 1, PIOR90 = 1)	
TO03	P125 (PIOR13 = 0)	P127 (PIOR13 = 1, PIOR90 = 0)	P62 (PIOR13 = 1, PIOR90 = 1)	
TO07	P120 (PIOR17 = 0)	P44 (PIOR17 = 1, PIOR90 = 0)	P63 (PIOR17 = 1, PIOR90 = 1)	

For the following, also note the pin functions used at the same time.

#### • SCK10, SI10, SO11 (CSI10, CSI11 input/output pin for serial array unit 1)

Pin Function	Assigned Port Name (Condition of PIORx Register)			
SCK10	P10 (PIOR42 = 0, PIOR91 = 0)	P76 (PIOR42 = 1)	P120 (PIOR42 = 0, PIOR91 = 1)	
SI10	P11 (PIOR42 = 0, PIOR91 = 0)	P75 (PIOR42 = 1)	P41 (PIOR42 = 0, PIOR91 = 1)	
SO10 Note	P12 (PIOR42 = 0) P74 (PI		)R42 = 1)	
SSI10 Note	P54 (PIOR42 = 0) P77 (PIC		)R42 = 1)	
SCK11 Note	P71 (PIC	PR43 = 0)	P153 (PIOR43 = 1)	
SI11 Note	P70 (PIC	PR43 = 0)	P152 (PIOR43 = 1)	
SO11	P72 (PIOR43 = 0, PIOR92 = 0)	P32 (PIOR43 = 0, PIOR92 = 1)	P151 (PIOR43 = 1)	
SSI11 Note	P73 (PIC	P150 (PIOR43 = 1)		

### • RXD1 (UART1 input pin for serial array unit 1)

Pin Function	Assigned Port Name (Condition of PIORx Register)		
RXD1	P11 (PIOR42 = 0, PIOR91 = 0)	P41 (PIOR42 = 0, PIOR91 = 1)	P75 (PIOR42 = 1)
TXD1 Note	P12 (PIOR42 = 0)		P74 (PIOR42 = 1)

#### LRXD1, LTXD1 (LIN/UART input/output pin for RLIN3 channel 1)

Pin Function	Assigned Port Name (Condition of PIORx Register)				
LRXD1	P11 (PIOR45 = 0)	P107 (PIOR45 = 1, PIOR93 = 0)	P125 (PIOR45 = 1, PIOR93 = 1)		
LTXD1	P10 (PIOR45 = 0)	P106 (PIOR45 = 1, PIOR93 = 0)	P120 (PIOR45 = 1, PIOR93 = 1)		

Note These pin functions are indicated to be used in combination with the corresponding pin functions.



# No.5: Correct the starting point of the arrow when restarting the operation.

Page: P.437

Operation is resumed.

#### Incorrect: Starting point of the arrow: TAU stop $\rightarrow$ Operation start

#### Figure 6-66. Operation Procedure When Delay Counter Function Is Used

	Software Operation	Hardware Status
TAU default setting		Power-off status (Clock supply is stopped and writing to each register is disabled.)
	Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.	Power-on status. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.	
Channel default setting	Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on). Sets timer mode register mn (TMRmn) (determines operation mode of channel). INTTMmn output delay is set to timer data register mn (TDRmn). Clears the TOEmn bit to 0 and stops operation of TOmn.	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	Sets the TSmn bit to 1. The TSmn bit automatically returns to 0 because it is a trigger bit.	TEmn = 1, and the TImn pin input valid edge detection wait status is set.
	Detects the TImn pin input valid edge.	Value of the TDRmn register is loaded to the timer count register mn (TCRmn).
During operation	Set value of the TDRmn register can be changed. The TCRmn register can always be read. The TSRmn register is not used.	The counter (TCRmn) counts down. When TCRmn counts down to 0000H, INTTMmn is output, and countin stops (which leaves TCRmn at 0000H) until the next TIm pin input.
Operation stop	The TTmn bit is set to 1. The TTmn bit automatically returns to 0 because it is a trigger bit.	TEmn = 0, and count operation stops. The TCRmn register holds count value and stops.
TAU stop	The TAUmEN bit of the PER0 register is cleared to 0.	Power-off status All circuits are initialized and SFR of each channel is also initialized.

**Remarks 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 7)

2. Channels 7 to 4 of unit 1 are not provided in the RL78/F23 products.



#### **Correct:** Starting point of the arrow: Operation stop $\rightarrow$ Operation start

Figure 6-66.	Operation Procedure When Delay Counter Function Is Used
--------------	---

	Software Operation	Hardware Status
TAU default setting		Power-off status (Clock supply is stopped and writing to each register is disabled.)
	Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.	Power-on status. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.	
Channel default setting	Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on). Sets timer mode register mn (TMRmn) (determines operation mode of channel). INTTMmn output delay is set to timer data register mn (TDRmn). Clears the TOEmn bit to 0 and stops operation of TOmn.	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	Sets the TSmn bit to 1. The TSmn bit automatically returns to 0 because it is a trigger bit.	TEmn = 1, and the TImn pin input valid edge detection wait status is set.
	Detects the TImn pin input valid edge.	Value of the TDRmn register is loaded to the timer count register mn (TCRmn).
During operation	Set value of the TDRmn register can be changed. The TCRmn register can always be read. The TSRmn register is not used.	The counter (TCRmn) counts down. When TCRmn counts down to 0000H, INTTMmn is output, and counting stops (which leaves TCRmn at 0000H) until the next TImpin input.
Operation stop	The TTmn bit is set to 1. The TTmn bit automatically returns to 0 because it is a trigger bit.	TEmn = 0, and count operation stops. The TCRmn register holds count value and stops.
TAU stop	The TAUmEN bit of the PER0 register is cleared to 0.	Power-off status All circuits are initialized and SFR of each channel is also initialized.

**Remarks 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 7)

2. Channels 7 to 4 of unit 1 are not provided in the RL78/F23 products.



## No.6: Change the explanation of RWAIT bit.

#### Page: P.671

# Correct: Add an explanation of when to use alarm interrupt to the explanation of RWAIT bit. (TN-RL\*-A0123A/E)

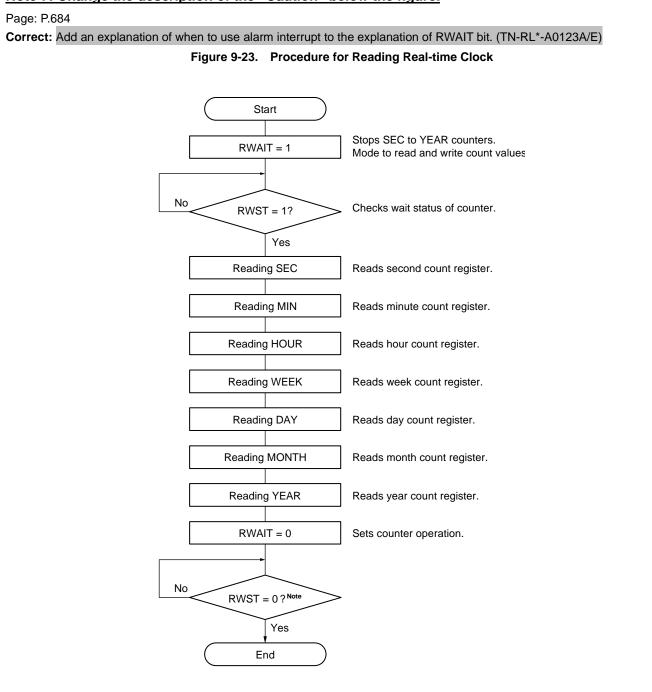
#### Figure 9-8. Format of Real-time Clock Control Register 1 (RTCC1) (2/2)

RWAIT	Wait control of real-time clock
0	Sets counter operation.
1	Stops SEC to YEAR counters. Mode to read or write counter value
This bit cont	rols the operation of the counter.
Be sure to w	rite "1" to it to read or write the counter value.
As the intern	al counter (16 bits) is continuing to run, complete reading or writing within one second and turn back to 0.
When RWAI Notes 1, 2	T = 1, it takes up to 1 operating clock ( $f_{RTC}$ ) until the counter value can be read or written (RWST = 1).
When readir	ng or writing to the counter is required while generation of the alarm interrupt is enabled, first set the
CT2 to CT0	bits to 010B (generating the constant-period interrupt once per 1 second).
Then, comp	lete the processing from setting the RWAIT bit to 1 to setting it to 0 before generation of the next
constant-per	iod interrupt.
When the in then counts	ternal counter (16 bits) overflowed while RWAIT = 1, it keeps the event of overflow until RWAIT = 0, up.

However, when it wrote a value to second count register, it will not keep the overflow event.



#### Note 7: Change the description of the "Caution" below the figure.

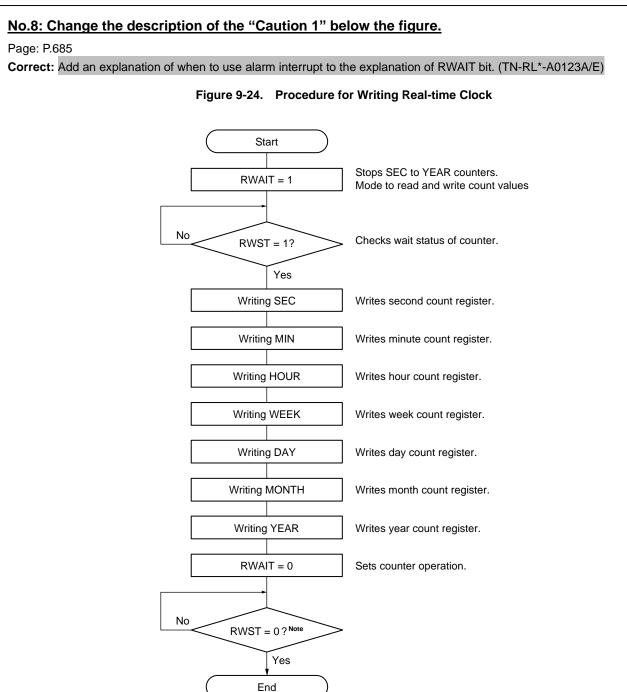


**Note** Be sure to confirm that RWST = 0 before setting HALT/STOP mode.

- Caution Complete the series of process of setting the RWAIT bit to 1 to clearing the RWAIT bit to 0 within 1 second. When reading to the counter is required while generation of the alarm interrupt is enabled, first set the CT2 to CT0 bits to 010B (generating the constant-period interrupt once per 1 second). Then, complete the processing from setting the RWAIT bit to 1 to setting it to 0 before generation of the next constant-period interrupt.
- Remark The second count register (SEC), minute count register (MIN), hour count register (HOUR), week count register (WEEK), day count register (DAY), month count register (MONTH), and year count register (YEAR) may be read in any sequence.

Also, it is not necessary to read all the registers, and only some of the registers may be read.





**Note** Be sure to confirm that RWST = 0 before setting HALT/STOP mode.

- Cautions 1. Complete the series of operations of setting the RWAIT bit to 1 to clearing the RWAIT bit to 0 within 1 second. When writing to the counter is required while generation of the alarm interrupt is enabled, first set the CT2 to CT0 bits to 010B (generating the constant-period interrupt once per 1 second). Then, complete the processing from setting the RWAIT bit to 1 to setting it to 0 before generation of the next constant-period interrupt.
  - 2. When changing the values of the SEC, MIN, HOUR, WEEK, DAY, MONTH, and YEAR register while the counter operates (RTCE = 1), rewrite the values of the MIN register after disabling interrupt servicing INTRTC by using the interrupt mask flag register. Furthermore, clear the WAFG, RIFG and RTCIF flags after rewriting the MIN register.
- Remark The second count register (SEC), minute count register (MIN), hour count register (HOUR), week count register (WEEK), day count register (DAY), month count register (MONTH), and year count register (YEAR) may be written in any sequence.

All the registers do not have to be set and only some registers may be written.



	Figu	re 12-16. l	Format of	A/D Disc	onnection I	Detection C	ontrol Registe	r (ADDISCR)	
Address	: F06BAH (Al				eset: 00H	R/W	U	· · · ·	
Symbol	7	6		5	4	3	2	1	0
DISCR	0	0		0		5	ADNDIS[4:0]	I	0
2.001.	<u> </u>	Ŭ		0					
	ADNDIS	4			Setting of r	orecharge/d	ischarge selection	on	
	0		charge se	lect	<b>e</b> etting er p	ge, a	ieenange eeneem		
	1		charge se						
	The ADNDI		•		e or discharg	e for the A/	D disconnection	detection ass	ist function.
					ADCSR.AD				
	ADNDIS[3	3:0]			Period	d of prechar	ge/discharge		
	0000B	Dis	ables the	disconne	ction detection	on assist fur	nction.		
	0001B	Se	tting prohil	bited.					
	0010B	2 0	ycles (2/A	DCLK)					
	0011B	3 0	ycles (3/A	DCLK)					
	0100B	4 c	ycles (4/A	DCLK)					
	0101B	5 c	ycles (5/A	DCLK)					
	0110B	6 0	ycles (6/A	DCLK)					
	0111B	7 c	ycles (7/A	DCLK)					
	1000B	8 0	ycles (8/A	DCLK)					
	1001B		ycles (9/A	,					
	1010B		cycles (10						
	1011B		cycles (11	,					
	1100B		cycles (12						
	1101B	13	cycles (13	3/ADCLK)					
	1110B		cycles (14						
	1111B		cycles (15	,					
	ADNDIS[3:0	0] bits sho	uld be set	to 0000B			sistance cannot 0000B or 0001		
	detection a	assistance	is enabl	ed, the		n detectior	n assistance fo		
	ADNDIS[3:0	0] bits sho	uld be set	while the	ADCSR.AD	ST bit is 0.			



/mbol DISCR	7			r reset: 00H	R/W			
DDISCR		6	5	4	3	2	1	0
L r	0	0	0			ADNDIS[4:0]		
Г								
	ADNDIS4			Setting of prech	arge/discharge	e selection Note 1		
	0	Discharge sel	ect					
	1	Precharge se	ect					
	The ADNDIS4	4 bit selects eith	er precharge or	discharge for th	e A/D disconne	ection detection a	assist function.	
	The ADNDIS4	4 bit should be s	et while the AD	CSR.ADST bit is	s 0.			
-								
	ADNDIS[3:0]			Period of p	precharge/discl	narge <sup>Note 2</sup>		
	0000B	Disables the	disconnection d	etection assist fu	inction.			
	0001B	Setting prohit	ited.					
	0010B	2 cycles (2/Al	DCLK)					
	0011B	3 cycles (3/Al	DCLK)					
	0100B	4 cycles (4/Al	DCLK)					
	0101B	5 cycles (5/Al	DCLK)					
	0110B	6 cycles (6/Al	DCLK)					
	0111B	7 cycles (7/Al	DCLK)					
	1000B	8 cycles (8/Al	DCLK)					
	1001B	9 cycles (9/Al	DCLK)					
	1010B	10 cycles (10	ADCLK)					
	1011B	11 cycles (11/	ADCLK)					
	1100B	12 cycles (12	ADCLK)					
	1101B	13 cycles (13	ADCLK)					
	1110B	14 cycles (14	ADCLK)					
	1111B	15 cycles (15	ADCLK)					
		f-diagnosis is us e set to 0000B.	sed, the discon	nection detectior	n assistance ca	annot be used. Ir	h that case, the	ADNDIS[3:0]
		enabled, the o		-		or 0001B, and annel-dedicated		

Notes 1. When the internal reference voltage is converted, A/D converter executes discharge automatically.

This operation is achieved by setting ADNDIS[4:0] to 0FH (15 cycles) automatically in setting ADEXICR.OCSA to 1. After executing discharge, the sampling will start.



# No.10: Correct the typo in the setting value of ADPAGE[3:0] bits when accessing ADSSTRL and ADSSTRO registers.

Page: P.737

Incorrect:

#### Figure 12-22. Format of A/D Converter Access Window Register (ADWINR)

ADPAGE[3:0]	A/D converter access window select bit
0000B	Read or write access is enabled for the registers of ADCSR, ADANSA0, ADANSA1, ADADS0, ADADS1, ADADC and ADCER.
0001B	Read or write access is enabled for the registers of ADSTRGR, ADEXICR, ADANSB0, ADANSB1, ADOCDR and ADRD.
0010B	Read access is enabled for the registers of ADDRy (y:0 to 7).
0011B	Read access is enabled for the registers of ADDRy (y:8 to 15).
0100B	Read access is enabled for the registers of ADDRy (y:16 to 23).
0101B	Read access is enabled for the registers of ADDRy (y:24 to 30).
0110B	Read or write access is enabled for the register of ADSHCR.
0111B	Read or write access is enabled for the register of ADDISCR.
1000B	Read or write access is enabled for the registers of ADGSPCR and ADHVREFCNT.
1001B	Read or write access is enabled for the registers of ADSSTRL and ADSSTRO.
1110B	Read or write access is enabled for the registers of ADSSTRn (n:0 to 15).
Other than	Setting prohibited.
above	
The ADPAGE b	its are used to select the A/D converter access window, and enable the register to be
read or written.	
Set this register	before accessing each A/D related register.

#### Correct:

## Figure 12-22. Format of A/D Converter Access Window Register (ADWINR)

ADPAGE[3:0]	A/D converter access window select bit
0000B	Read or write access is enabled for the registers of ADCSR, ADANSA0, ADANSA1, ADADS0, ADADS1, ADADC and ADCER.
0001B	Read or write access is enabled for the registers of ADSTRGR, ADEXICR, ADANSB0, ADANSB1, ADOCDR and ADRD.
0010B	Read access is enabled for the registers of ADDRy (y:0 to 7).
0011B	Read access is enabled for the registers of ADDRy (y:8 to 15).
0100B	Read access is enabled for the registers of ADDRy (y:16 to 23).
0101B	Read access is enabled for the registers of ADDRy (y:24 to 30).
0110B	Read or write access is enabled for the register of ADSHCR.
0111B	Read or write access is enabled for the register of ADDISCR.
1000B	Read or write access is enabled for the registers of ADGSPCR and ADHVREFCNT.
1101B	Read or write access is enabled for the registers of ADSSTRL and ADSSTRO.
1110B	Read or write access is enabled for the registers of ADSSTRn (n:0 to 15).
Other than	Setting prohibited.
above	
The ADPAGE b read or written.	its are used to select the A/D converter access window, and enable the register to be
Set this register	before accessing each A/D related register.



No.11: Correct a typo in Note 5 on the CSTEN bit.
---

# Page: P.784 Incorrect:

# Figure 14-3. Format of Comparator Control Register (CMPCTL)

Symbol	<7>	6	5	4	3	2	<1>	0
CMPCTL	HCMPON	CDFS1	CDFS0	CEGN	CEGP	CSTEN	COE	CINV
	CSTEN		ST	OP mode rele	ase enable <sup>Notes</sup>	5, 6		R/W
	0	Releasing S	OP mode by co	omparator inte	rrupt disabled			R/W
	1	Releasing S	OP mode by c	omparator inte	rrupt enabled			
	To enable rele CDFS0, and C Fi	CINV to 00B (	-	t used).				set bits C
(	CDFS0, and C Fi	CINV to 00B (	noise filter no	t used).				set bits C
( t:	CDFS0, and C Fi	CINV to 00B (	noise filter no	t used).				set bits C
( <b>t:</b> ddress: F02	CDFS0, and C Fi 2A0H After	CINV to 00B ( gure 14-3. reset: 00H	noise filter no Format of Co	t used). omparator C	ontrol Regis	ter (CMPCTL	)	
( <b>t:</b> ddress: F02 Symbol	CDFS0, and C Fi 2A0H After <7>	CINV to 00B ( gure 14-3. reset: 00H 6	Format of Co 5 CDFS0	t used). omparator C 4 CEGN	ontrol Regis	2 CSTEN	) <1>	0
( <b>t:</b> ddress: F02 Symbol	CDFS0, and C Fi 2A0H After <7> HCMPON	CINV to 00B ( gure 14-3. reset: 00H 6 CDFS1	Format of Co 5 CDFS0	t used). omparator C 4 CEGN OP mode relea	3 CEGP ase enable <sup>Notes</sup>	2 CSTEN	) <1>	0 CINV

# No.12: Add "Reception end interrupt" to [Interrupt function].

# Page: P.798

# Incorrect:

### 15.1.2 UART (UART0, UART1)

[Interrupt function]

• Transfer end interrupt/buffer empty interrupt

Correct:

# 15.1.2 UART (UART0, UART1)

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt
- Reception end interrupt

-----



Г

ge: P.815 correct:																
correct:		<b>F</b> laun	. 45 0			f Carla		Clear	<b>T</b> u:	. Deel	-1					
		Figui	re 15-9	). FOI	rmat o	r Seria	ii Fiag	Clear	irigge	r Regi	ster m	n (Sik	(mn)			
Address: F0	104H, F(	0105H (	(SIR00)	, F0106	6H, F01	07H (SI	IR01),	Afte	er reset:	: 0000H	I	R/W				
F0	144H, F(	0145H (	(SIR10)	, F0146	6H, F01	47H (SI	IR11)									
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SIRmn	0	0	0	0	0	0	0	0	0	0	0	0	0	FECT	PEC	OVC
															-	Tmn
Cautions	2. Us in	e the S this re	SIRmn	regist is cl	ter to c eared,	clear o								If the e		ag no
Cautions	2. Us in	e the S this re	SIRmn egister	regist is cl	ter to c eared,	clear o								If the e	rror fla	ag no
	2. Us in	e the S this re aring	SIRmn egister this er	regist is cl ror fla	ter to c eared, g.	the f	lag ma	ay be	erase	d whei	nan (	error i	s dete	If the e	rror fla	ag no
	2. Us in	e the S this re aring	SIRmn egister this er	regist is cl ror fla	ter to c eared, g.	the f	lag ma	ay be	erase		nan (	error i	s dete	If the e	rror fla	ag no
orrect: Address: F0	2. Us in cle	e the S this re aring Figur	SIRmn egister this er re 15-9 (SIR00)	regist ris cl ror fla ). For	ter to c eared, g. rmat o	the f f Seria	lag ma al Flag IR01),	ay be Clear	erase Trigge	d whei	n an o ster m	error i	s dete	If the e	rror fla	ag no
orrect: Address: F0	2. Us in cle	e the S this re aring Figur	SIRmn egister this er re 15-9 (SIR00)	regist is cl ror fla ). For	ter to c eared, g. rmat o	the f f Seria	lag ma al Flag IR01),	ay be Clear	erase Trigge	d whei r Regi	n an o ster m	error i n (SIR	s dete	If the e	rror fla	ag no
p <b>rrect:</b> Address: F0 F0	2. Us in cle 104H, F( 144H, F(	e the S this re aring Figur 0105H ( 0145H (	SIRmn egister this er re 15-9 (SIR00) (SIR10)	regist r is cl ror fla . For , F0106 , F0146	ter to c eared, g. rmat o SH, F01 SH, F01	the f f Seria	<b>lag m</b> a <b>hi Flag</b> IR01), IR11)	ay be Clear Afte	erased Trigge	d whei r Regi : 0000H	n an o ster m	error i n (SIR RW	s dete amn)	If the e	rror fla	ag no eadin

 Use the SIRmn register to clear only the error flag set in the SSRmn register. If the error flag not set in this register is cleared, the flag may be erased when an error is detected from reading to clearing this error flag.



Page: P.816																
ncorrect:																
		Fig	jure 1	5-10.	Forma	at of S	erial S	tatus F	Regist	er mn	(SSRm	n) (1/2	2)			
Address: F0	100H, F(	0101H (	SSR00	), F010	2H, F01	103H (S	SR01),	Af	ter rese	et: 0000	н	R				
	140H, F0															
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSRmn	0	0	0	0	0	0	0	0	0	TSF mn	BFF mn	0	0	FEF mn	PEF mn	OVF mn
•			•••										, ,			
Caution	If data					-						ansmit	recei	ve dat	a stor	ed in
	regist	, 15 ui	Scara			cirun		<b></b>	uu <del></del>	15 461	colca.					
Correct:																
		Fig	jure 1	5-10.	Forma	at of S	erial S	tatus F	Regist	er mn	(SSRm	nn) (1/2	2)			
Address: F0	100H F(	)101H (	SSR00	) F010	2H F01	03H (S	SR01)	Af	ter rese	et: 0000	н	R				
	140H, F(							7.0				i.				
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSRmn	0	0	0	0	0	0	0	0	0	TSF	BFF	0	0	FEF	PEF	OVF
										mn	mn			mn	mn	mn
Caution <u>Io.15: Add t</u> <u>exampl</u>	registe <u>he des</u>	er is di script	scard	ed and of ser	ian ov ial da	verrun ata in	error ( put/o	OVFm	ın = 1)	is det	the tra ected.					
<b>lo.15: Add t</b> <u>exampl</u> Page: P.829	registe <u>he des</u>	er is di script	scard	ed and of ser	ian ov ial da	verrun ata in	error ( put/o	OVFm	ın = 1)	is det	the tra ected.					
lo.15: Add t <u>exampl</u> Page: P.829 Correct:	registe <u>he des</u> e whe	er is di <u>script</u> n usir	iscarde ion o ng the	ed and o <u>f ser</u> e SDA	ian ov <u>ial da</u> Vmn fi	errun ata in unctio	error ( <u>put/o</u> on.	OVFm	ın = 1)	is det	the tra ected.					
No.15: Add ti <u>exampl</u> Page: P.829 Correct: 5.3.19 Port M	registe <u>he des</u> le whe ode Reg	er is di <u>script</u> n usir gisters	ion o ng the	ed and o <u>f ser</u> e SDA	d an ov <u>ial da</u> <u>vmn f</u> u to PM	verrun a <u>ta in</u> unctio 7, PM1	error ( <u>put/o</u> on. 2)	OVFm utput	ın = 1) <u>(SD</u> /	is det	the tra ected.					
lo.15: Add t <u>exampl</u> Page: P.829 Correct:	registe he des e whe ode Reg	er is di script n usir gisters	ion o ng the s (PM1 utput c	ed and of ser e SDA	to PM	verrun a <u>ta in</u> unctio 7, PM1 7, and	error ( <u>put/o</u> on. (2)	OVFm utput	n = 1) (SD/	is det Amn)	the tra ected. to th	e tex	t. Als	so ad	<u>d the</u>	e sett
No.15: Add ti exampl Page: P.829 Correct: 5.3.19 Port M These regist When using port mode re	registe he des e whe ode Reg ers set i the seri egister (	er is di script n usir gisters input/o al data	iscard ion o ng the s (PM1 utput c i outpu	ed and of ser e SDA , PM3 of ports at or se	to PM to In a to rial clo	verrun a <u>ta in</u> unctio 7, PM1 7, and ck out <sub>i</sub>	error ( put/o on. 12) 12 in put, or	OVFm utput 1-bit ur when u	n = 1) (SD) hits. using s	is det <u>Amn)</u> serial d	the tra ected. to th	<mark>e tex</mark> ut/outp	t. Als	oAmn),	<u>d the</u> set th	e sett
<b>No.15: Add t</b> <u>exampl</u> Page: P.829 Correct: <b>5.3.19 Port M</b> These regist When using port mode re each port to	registe he des e whe ers set i the seri egister ( 1.	er is di script n usir gisters input/o al data (PMmn	ion o ng the s (PM1 utput c i outpu ) corre	ed and of ser e SDA , PM3 of ports to r se espond	to PM rial clo to PM rial clo ling to	verrun a <u>ta in</u> unctio 7, PM1 7, and ck out each p	error ( put/o on. 12) 12 in put, or port to	OVFm utput 1-bit ur when u 0. And	n = 1) (SD/ hits. using s I set th	is det Amn) serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
No.15: Add ti exampl Page: P.829 Correct: 5.3.19 Port M These regist When using port mode re each port to Also, when u	registe he des e when ode Reg ers set i the seri egister ( 1. using the	er is di script n usir gisters input/o al data (PMmn e serial	ion o ng the s (PM1 utput o n output ) corre	ed and <u>of ser</u> <u>e SDA</u> of ports tt or se espond nput o	to PM ial da mn fi to PM i1, 3 to rial clo ling to r serial	verrun ata in unctio 7, PM1 7, and ck out each p clock	error ( put/o on. 12) 12 in put, or port to	OVFm utput 1-bit ur when u 0. And	n = 1) (SD/ hits. using s I set th	is det Amn) serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>No.15: Add t</b> <u>exampl</u> Page: P.829 Correct: <b>5.3.19 Port M</b> These regist When using port mode re each port to	registe he des e when ode Reg ers set i the seri egister ( 1. using the	er is di script n usir gisters input/o al data (PMmn e serial	ion o ng the s (PM1 utput o n output ) corre	ed and <u>of ser</u> <u>e SDA</u> of ports tt or se espond nput o	to PM ial da mn fi to PM i1, 3 to rial clo ling to r serial	verrun ata in unctio 7, PM1 7, and ck out each p clock	error ( put/o on. 2) 1 12 in put, or port to	OVFm utput 1-bit ur when u 0. And	n = 1) (SD/ hits. using s I set th	is det Amn) serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
No.15: Add ti exampl Page: P.829 Correct: 5.3.19 Port M These regist When using port mode re each port to Also, when u	registe he des e whe ers set i the seri egister ( 1. using the the bit in	er is di script n usir gisters input/o al data (PMmn e serial n the P	ion o ng the s (PM1 utput c i outpu ) corre data i mn bit	ed and of ser e SDA of ports to r se espond nput o	to PM to PM and to PM rial clo ling to r serial to or	<b>a<u>ta</u> in <u>unctio</u> 7, PM1 7, and ck out each p clock 1.</b>	error ( put/o on. 12 in put, or port to input, s	OVFm utput 1-bit ur when u 0. And et the	n <b>n = 1)</b> (SD) hits. Jusing s I set th PMmn	is det Amn) serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>No.15: Add the example</b> Page: P.829 Correct: <b>5.3.19 Port M</b> These regist When using port mode re each port to Also, when u At this time,	registe he des e whe ers set i the seri egister ( 1. using the the bit in	er is di script n usir gisters input/o al data (PMmn e serial n the P n using	ion o ng the s (PM1 utput o i output ) corre data i mn bit mn bit	ed and <u>of ser</u> <u>e SDA</u> , PM3 of ports it or se espond nput o : may b SO10/	to PM to PM and to PM rial clo ling to r serial to or	<b>a<u>ta</u> in <u>unctio</u> 7, PM1 7, and ck out each p clock 1.</b>	error ( put/o on. 12 in put, or port to input, s	OVFm utput 1-bit ur when u 0. And et the	n <b>n = 1)</b> (SD) hits. Jusing s I set th PMmn	is det Amn) serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>No.15: Add the example</b> Page: P.829 Correct: <b>5.3.19 Port M</b> These regist When using port mode re each port to Also, when u At this time,	registe he des e when ode Reg ers set i the seri egister ( 1. using the the bit in 1: Whe	er is di script n usir gisters input/o al data (PMmn e serial n the P n using e PM12	ion o ng the s (PM1 utput o outpu ) corre data i mn bit g P12/s 2 bit to	ed and <u>of ser</u> <u>e SDA</u> of ports to r se espond nput o : may b <b>SO10</b> / <sup>7</sup> 0.	to PM to PM a1, 3 to rial clo ling to r serial be 0 or TXD1 f	<b>ata in</b> <b>ata in</b> <b>unctio</b> <b>7, PM1</b> 7, and ck out each p clock i 1. or seria	error ( put/o on. 12 in put, or port to input, s	OVFm utput 1-bit ur when u 0. And et the	n <b>n = 1)</b> (SD) hits. Jusing s I set th PMmn	is det Amn) serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>No.15: Add the example</b> Page: P.829 Correct: <b>5.3.19 Port M</b> These regist When using port mode re each port to Also, when u At this time,	register he des e when ode Reg ers set if the seri egister ( 1. using the the bit in 1: Whe Set the Set the	er is di <u>script</u> <u>n usir</u> gisters input/o al data (PMmn e serial n the P n using e PM12 e P12 b	ion o ng the s (PM1 utput o a output ) corre d data i mn bit g P12/s 2 bit to bit of th	ed and <u>of ser</u> <u>e SDA</u> , PM3 of ports it or se espond nput o : may b SO10/ <sup>-</sup> 0. ne P1 r	to PM to PM to PM to I, 3 to rial clo ling to r serial be 0 or TXD1 f egister	<b>ata in</b> <b>unctio</b> <b>7, PM1</b> <b>7, and</b> ck out each p clock 1. or seria to 1.	error ( put/o on. 12) 112 in put, or port to input, s al data	OVFm utput 1-bit ur when u 0. And et the output	n = 1) (SD/ hits. using s I set th PMmn	is det Amn) Serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>Jo.15: Add th</b> <b>exampl</b> Page: P.829 <b>Correct:</b> <b>5.3.19 Port M</b> These regist When using port mode re each port to Also, when u At this time, Example	register he des e when ode Reg ers set if the seri egister ( 1. using the the bit in 1: Whe Set the Set the	er is di script n usir gisters input/o al data (PMmn e serial n the P n using e PM12 e P12 b n using	ion o ng the s (PM1 utput o a outpu ) corre d data i mn bit g P12/s 2 bit to bit of th g P11/s	ed and <u>of ser</u> <u>a SDA</u> of ports to r se aspond nput o may b <b>SO10</b> /7 0. ne P1 r SI10/S	to PM to PM to PM to I, 3 to rial clo ling to r serial be 0 or TXD1 f egister	<b>ata in</b> <b>unctio</b> <b>7, PM1</b> <b>7, and</b> ck out each p clock 1. or seria to 1.	error ( put/o on. 12) 112 in put, or port to input, s al data	OVFm utput 1-bit ur when u 0. And et the output	n = 1) (SD/ hits. using s I set th PMmn	is det Amn) Serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>Jo.15: Add th</b> <b>exampl</b> Page: P.829 <b>Correct:</b> <b>5.3.19 Port M</b> These regist When using port mode re each port to Also, when u At this time, Example	registe he des e when ode Reg ers set i the seri egister ( 1. using the the bit in 1: Whe Set the Set the 2: Whe	er is di script n usir gisters input/o al data (PMmn e serial n the P n using e PM12 e P12 b n using e PM11	ion o ng the s (PM1 utput o output ) corre d data i mn bit g P12/s 2 bit to bit of th g P11/s bit to	ed and <u>of ser</u> <u>e SDA</u> , PM3 of ports to r se espond nput o : may b SO10/T 0. ne P1 r SI10/S 0.	to PM ial da ing to r serial be 0 or TXD1 f egister	<b>ata in</b> <b>unctio</b> <b>7, PM1</b> 7, and ck out each p clock i 1. or seria to 1. RXD1 f	error ( put/o on. 12) 112 in put, or port to input, s al data	OVFm utput 1-bit ur when u 0. And et the output	n = 1) (SD/ hits. using s I set th PMmn	is det Amn) Serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
Ao.15: Add the example of the exampl	register he des e when ode Reg ers set i the seri egister ( 1. using the the bit in 1: Whe Set the Set the Set the Set the Set the	er is di script n usir gisters input/o al data (PMmn e serial n the P n using e PM12 e P12 b n using e PM11 e P11 b	ion o ng the s (PM1 utput o output ou	ed and of ser e SDA of ports to r se espond nput o may b SO10/7 0. ne P1 r SI10/SI 0. ne P1 r	to PM to PM a1, 3 to rial clo ling to r serial be 0 or TXD1 f egister	<b>ata in</b> <b>unctio</b> <b>7, PM1</b> 7, and ck out each p clock i 1. or seria to 1. RXD1 1 to 1.	error ( put/o on. (2) (12) (12) (12) (12) (12) (12) (12)	OVFm <u>utput</u> 1-bit ur when u 0. And et the output	in = 1) (SD/ hits. using s I set th PMmn	is det Amn) Serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett
<b>Jo.15: Add th</b> <b>exampl</b> Page: P.829 <b>Correct:</b> <b>5.3.19 Port M</b> These regist When using port mode re each port to Also, when u At this time, Example	register he des e when ode Reg ers set i the seri egister ( 1. using the the bit in 1: Whe Set the Set the Set the Set the Set the	er is di <u>script</u> <u>n usir</u> gisters input/o al data (PMmn e serial n the P n using e PM12 e PM12 e PM12 e PM11 e P11 b n using m	ion o ng the s (PM1 utput o output ) corre d data i mn bit g P12/3 bit to bit of th bit to pit of th g P16/3	ed and of ser e SDA b SDA of ports t or se espond nput o may b SO10/T 0. ne P1 r SI10/S 0. ne P1 r SI10/S	to PM to PM a1, 3 to rial clo ling to r serial be 0 or TXD1 f egister	<b>ata in</b> <b>unctio</b> <b>7, PM1</b> 7, and ck out each p clock i 1. or seria to 1. RXD1 1 to 1.	error ( put/o on. (2) (12) (12) (12) (12) (12) (12) (12)	OVFm <u>utput</u> 1-bit ur when u 0. And et the output	in = 1) (SD/ hits. using s I set th PMmn	is det Amn) Serial d ne bit i	the tra ected. to th ata inp n the p	e tex ut/outp ort reg	<b>t. Als</b> but (SE gister (	So ad DAmn), (Pmn) (	<b>d the</b> set the	e sett



#### No.16: Correct a typo in SOmn output. Page: P.901 Incorrect: There is an error in the description of the SOmn pin output. The correct statement is that it changes according to the input of the SSImn pin. Figure 15-75. Slave Select Input Function Timing Diagram DAPmn = 0 Transmit data is set BFFmn TSFmn SSEmn SCKmn (CKPmn = 0) SImn (bit6) bit5 (bit4) (bit3) (bit2 (bit1) (bit0 bit7 Sampling timing 4 ŧ ŧ ŧ f f f f SOnm х bit7Xbit6Xbit5Xbit4Xbit3Xbit2Xbit1 bit0 SSImn While SSImn is at high level, transmission is not performed even if the falling edge of SCKmn (serial clock) arrives, and neither is receive data sampled in synchronization with the rising edge. When SSImn goes to low level, data is output (shifted) in synchronization with the falling edge of the serial clock and a reception operation is performed in synchronization with the rising edge. DAPmn = 1 Transmit data is set BFFmn TSFmn SSEmn SCKmn (CKPmn = 0)bit1 bit0 SImn bit6 bit5 bit4 bit3 bit2 Sampling timing f f f f ł f SOmn bit7 bit6 vit5 bit4 bit3 bit2 bit1 bit0 SSImn If DAPmn = 1, when transmit data is set while SSImn is at high level, the first data (bit 7) is output to the data output. However, no shift operation is performed even if the rising edge of SCKmn (serial clock) arrives, and neither is receive data sampled in synchronization with the falling edge. When SSImn goes to low level, data is output (shifted) in synchronization with the next rising edge and a reception operation is performed in synchronization with the falling edge.

RENESAS

	Figure 15-75.	Slave Select Inpu	It Function Timing Diagram	•
DAPmn = 0 Trai	nsmit data is set			
5	¥			
BFFmn	]			
TSFmn				
SSEmn				
SCKmn (CKPmn = 0)				
SImn			bit7 bit6 bit5 bit4 bit3 bit2	bit1 bit0
Sampling timing	<b>† † † †</b>			
SOmn	Hi-Z		<u>X bit7 bit6 bit5 bit4 bit3 bit2</u>	X bit1 X bit0 X Hi-Z
SSImn				
is the rece When SS	ive data sampled in sy mn goes to low level,	nchronization with t	he rising edge.	serial clock) arrives, and neit e falling edge of the serial clo
is the rece When SS and a rece	ive data sampled in sy mn goes to low level,	nchronization with t	he rising edge. ed) in synchronization with th	
is the rece When SS and a rece DAPmn = 1 Tran BFFmn	ive data sampled in sy mn goes to low level, a ption operation is perfe	nchronization with t	he rising edge. ed) in synchronization with th	
is the rece When SS and a rece DAPmn = 1 Tran BFFmn TSFmn	ive data sampled in sy mn goes to low level, a ption operation is perfe	nchronization with t	he rising edge. ed) in synchronization with th	
is the rece When SS and a rece DAPmn = 1 Trar BFFmn TSFmn SSEmn	ive data sampled in sy mn goes to low level, a ption operation is perfe	nchronization with t	he rising edge. ed) in synchronization with th	
is the rece When SS and a rece DAPmn = 1 Tran BFFmn TSFmn	ive data sampled in sy mn goes to low level, a ption operation is perfe	nchronization with t	he rising edge. ed) in synchronization with th	
is the rece When SS and a rece DAPmn = 1 Trar BFFmn TSFmn SSEmn (CKPmn = 0) SImn	ive data sampled in sy mn goes to low level, a ption operation is perfe	nchronization with t	he rising edge. ed) in synchronization with th	e falling edge of the serial clo
is the rece When SS and a rece DAPmn = 1 Trar BFFmn TSFmn SSEmn SCKmn (CKPmn = 0) SImn Sampling timing	ive data sampled in sy mn goes to low level, o eption operation is performant ismit data is set	nchronization with t	he rising edge. ed) in synchronization with th ation with the rising edge.	e falling edge of the serial clo
is the rece When SS and a rece DAPmn = 1 Trar BFFmn TSFmn SSEmn (CKPmn = 0) SImn Sampling timing SOmn	ive data sampled in sy mn goes to low level, a ption operation is perfe	nchronization with t	he rising edge. ed) in synchronization with th ation with the rising edge.	e falling edge of the serial clo
is the rece When SS and a rece DAPmn = 1 Trar BFFmn TSFmn SSEmn SCKmn (CKPmn = 0) SImn Sampling timing	ive data sampled in sy mn goes to low level, o eption operation is performant ismit data is set	nchronization with t	he rising edge. ed) in synchronization with th ation with the rising edge.	e falling edge of the serial clo
is the rece When SS and a rece DAPmn = 1 Trar BFFmn TSFmn SSEmn (CKPmn = 0) SImn Sampling timing SOmn	ive data sampled in sy mn goes to low level, o eption operation is performant ismit data is set	nchronization with t	he rising edge. ed) in synchronization with th ration with the rising edge.	e falling edge of the serial clo

and a reception operation is performed in synchronization with the falling edge.



# No.17: Add slave selection signal output pin to the figure.

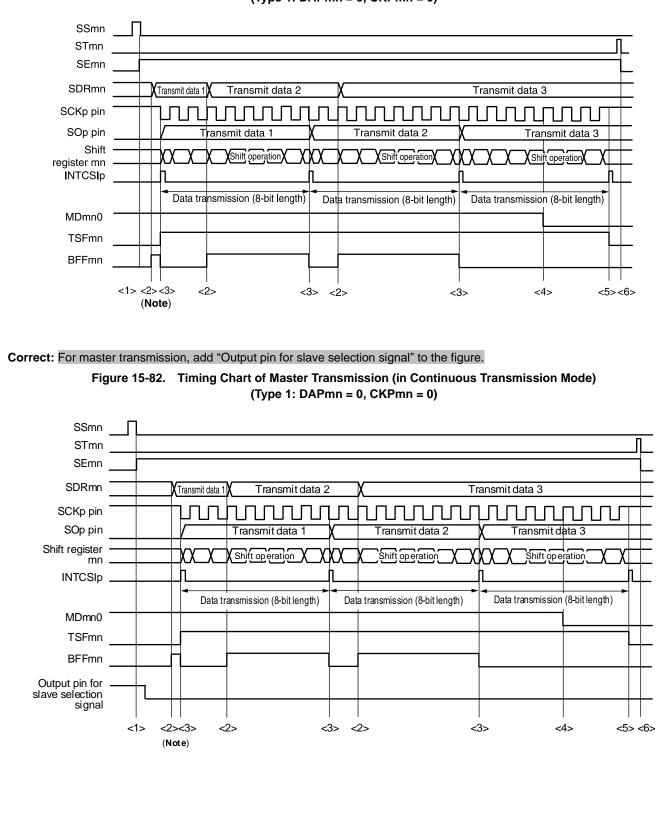
	Figure 15-80. Timing Chart of Master Transmission (in Single-Transmission Mode) (Type 1: DAPmn = 0, CKPmn = 0)
SSmn _	
STmn _	
SEmn _	
SDRmn	Transmit data 1 Transmit data 2 Transmit data 3
SCKp pin — SOp pin	Transmit data 1
Shift —	Transmit data 1     Transmit data 2     Transmit data 3       VXX_Shift operation     VXX_Shift operation     VX
register mn INTCSIp	
TSFmn	Data transmission (8-bit length)
orrect: For mas	ster transmission, add "Output pin for slave selection signal" to the figure.
orrect. For mas	Figure 15-80. Timing Chart of Master Transmission (in Single-Transmission Mode)
	(Type 1: DAPmn = 0, CKPmn = 0)
SSmn STmn	
SEmn	
SDRmn	Transmit data 1 X Transmit data 2 X Transmit data 3
SCKp pin	
SOp pin	Transmit data 1 Transmit data 2 Transmit data 3
Shift register mn	XXX         Shift operation         XXX         Shif
INTCSIp	
TSFmn	Data transmission (8-bit length)       Data transmission (8-bit length)         Data transmission (8-bit length)       Data transmission (8-bit length)
Output pin for slave selection	
signal	



### No.18: Add slave selection signal output pin to the figure.



#### Figure 15-82. Timing Chart of Master Transmission (in Continuous Transmission Mode) (Type 1: DAPmn = 0, CKPmn = 0)

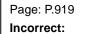




#### No.19: Add the slave selection signal output pin to the figure. Page: P.917 Incorrect: Figure 15-88. Timing Chart of Master Reception (in Single-Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0) SSmn STmr SEmn Receive data 3 Receive data 1 -Receive data 2 SDRmn Dummy data Dummy data for reception Dummy data ∕₩rite <sup>4</sup>Write ▲Read \_4Write ▲ Read Read A SCKp pin 11 SIp pin Receive data 1 Receive data 2 Receive data 3 Shift Reception & shift operation Reception & shift operation Reception & shift operation register mn INTCSIp Data reception (8-bit length) Data reception (8-bit length) Data reception (8-bit length) TSFmn Correct: For master reception, add "Output pin for slave selection signal" to the figure. Figure 15-88. Timing Chart of Master Reception (in Single-Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0) SSmn STmn SEmn Receive data 2 Receive data 1 Receive data 3 Dummy data for reception Dummy data SDRmn Dummy data A Write ▲ Read A Write ▲Read Write Read A SCKp pin SIp pin Receive data 1 Receive data 2 Receive data 3 Shift register Reception & shift operation Reception & shift operation Reception & shift operation mn **INTCSIp** Data reception (8-bit length) Data reception (8-bit length) Data reception (8-bit length) TSFmn Output pin for slave selection signal



#### No.20: Add the slave selection signal output pin to the figure.



TSFmn BFFmn

<2> <3>

<1>

<2>

<3>

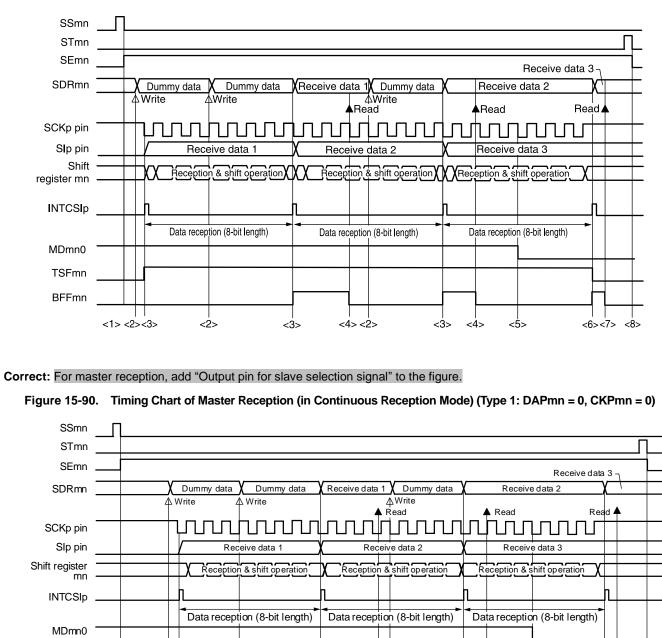
<4><2>

<3> <4>

<5>

Output pin for slave selection signal

#### Figure 15-90. Timing Chart of Master Reception (in Continuous Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)



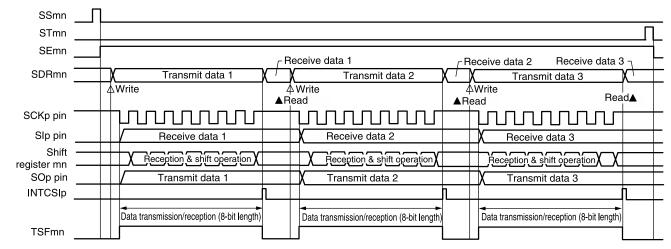
<6><7> <8>

#### No.21: Add the slave selection signal output pin to the figure.

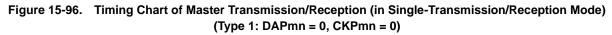


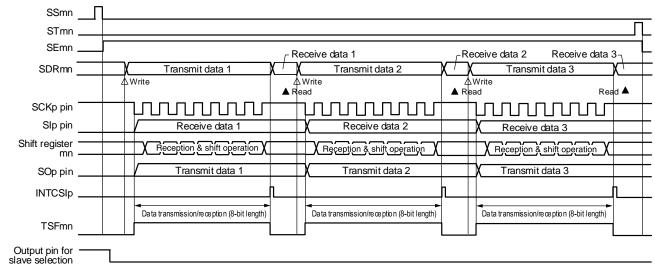


#### Figure 15-96. Timing Chart of Master Transmission/Reception (in Single-Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)



#### Correct: For master transmission/reception, add "Output pin for slave selection signal" to the figure.



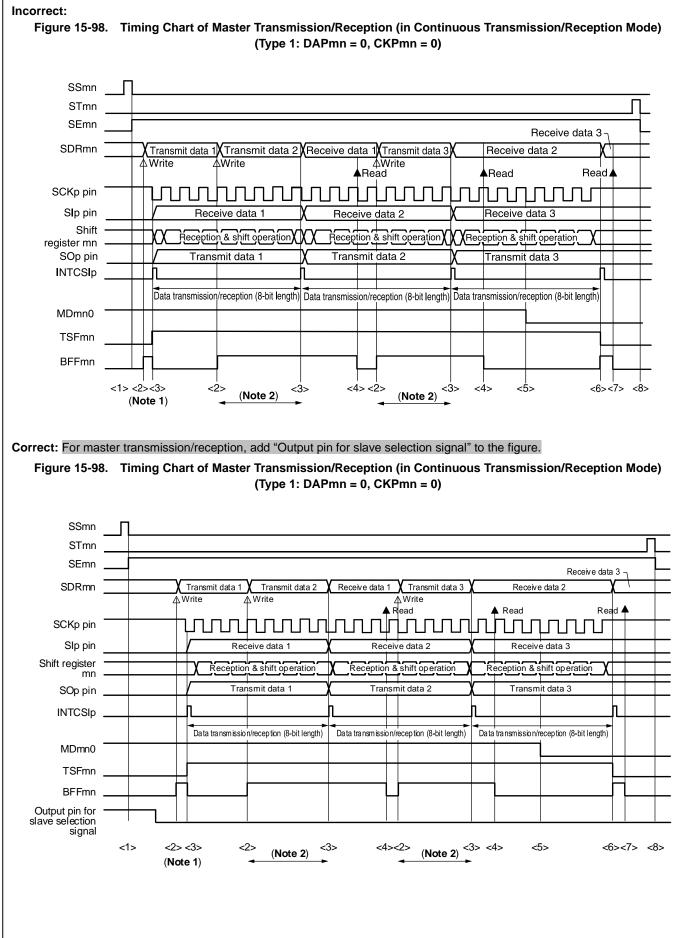


ave selection signal



### No.22: Add the slave selection signal output pin to the figure.





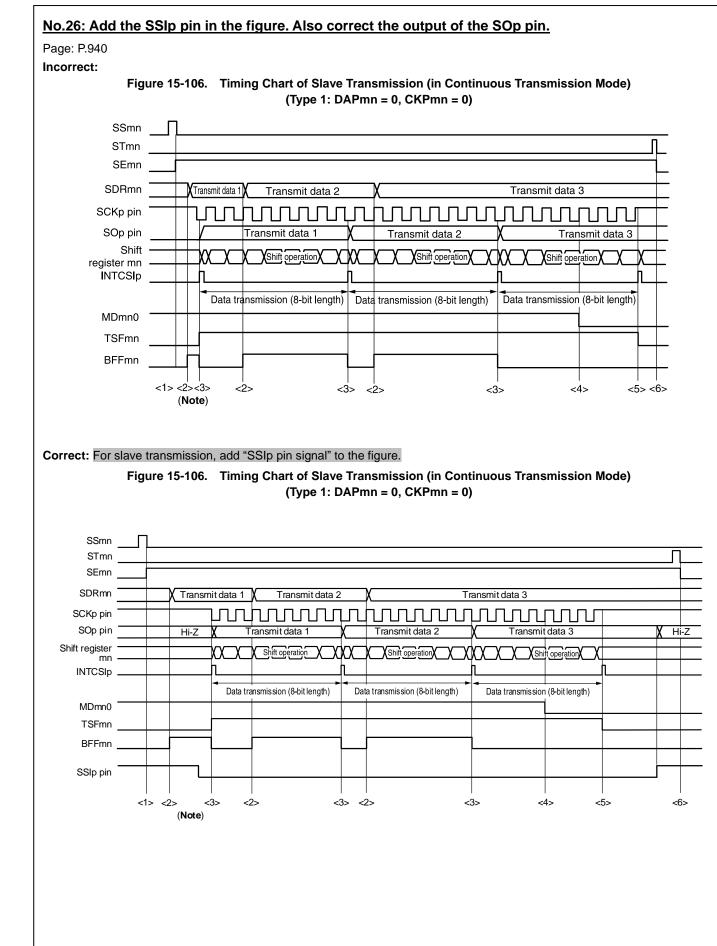
		regis	ster s	ettin	g oro	ier in	the f	igure	in tr	e sar	ne w	ay as	s III u		gure	13-10
e: P.934 r <b>rect:</b>																
Fi	gure 1	15-100	). Ex	ample				egiste , CSI1				smiss	ion of	f SPI	Functi	on
(d) Serial	l slave	) sele	ct ena	ble re	gister	<u>m (S</u>	SEm)								nd SSI e mode	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSEm1 0/1	SSEm 0/1
•					-	-							-			
						0	1 41			- 4	- 4 - 1					
(e) Serial	1 <b>OUTP</b> 15	ut reg 14	13	n (SO 12	<b>m)</b> 11	<b>Sets</b> (	oniy ti 9		7 of th	e targ 6	<b>ет спа</b> 5	innei. 4	3	2	1	0
SOm	15	14	13	12	11	10		o CKOm0	_	0	5	4	3	2	SOm1	SOm
3011	0	0	0	0	0	0	×	×	0	0	0	0	0	0	0/1	0/1
r	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	
-			,													0/1
														•		0/1
(g) Serial	l chan	nel st	art re	gister	m (SS	Sm)	. Sets	only t	he bit	s of th	ne tarç	jet ch	annel	to 1.		0/1
(g) Serial	<b>I chan</b> 15	nel st	art reg	gister 12	<b>m (S</b> \$ 11	<b>Sm)</b> 10	. <b>Sets</b> 9	only t	he bit 7	s of th	ne tarç 5	<b>jet ch</b> 4	annel 3	<b>to 1</b> .	1	0/1
<b>(g) Seria</b> l SSm				-	-	-		-				-			1 SSm1 0/1	0/1 0 SSm( 0/1



Correct:																	
	Figure	15-100	). Ex	ample				-				smiss	ion of	f SPI I	Functi	on	
					(C	5100,	CSI01	, CSN	0, 65	111) (2	(12)						
(d) Se	rial outp	out reg	ister	m (SO	m)	Sets	only ti	ne bits	s of th	e targ	et cha	innel.					
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SOm	0	0	0	0	0	0	CKOm1 ×	CKOm0 ×	0	0	0	0	0	0	SOm1 0/1	SOm0 0/1	
(e) Serial output enable register m (SOEm) Sets only the bits of the target channel to 1.																	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SOEn	n O	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	SOEm0 0/1	
(f) Se	rial slav	e sele	ct ena	ble re	gister	' m (S	SEm)								nd SSI mode	11 pin	
	15	14	13	12	11	10	9	8	puts נ 7	6	5	4	3	2	1	<b>.</b> 0	
SSEn		0	0	0	0	0	0	0	0	0	0	0	0	0	SSEm1 0/1		
						_			_								
(a) Se	rial char	nnel si	tart re	aister	m (SS	Sm)	Sets	only t	he bit	s of th	ne taro	iet ch	annel	to 1.			
(9) 00	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
00															SSm1	SSm0	
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1	0/1	
No.24: Corr	ect the	cont	ent i	า "Re	mark	: 1" b	elow	the f	iaure								
Page: P.936										-							
Incorrect:																	
			Figu	ure 15 <sup>.</sup>	-102.	Proc	edure	for St	oppin	ig Slav	ve Tra	nsmis	sion				
Remar															-		e-set serial
	0	utput r	egiste	rm (S	Om) (:	see Fi	gure 1	5-103	Pro	cedur	e for F	Resum	ning S	lave	Fransr	nissio	ו).
Correct:																	
			Figu	ıre 15.	-102	Proc	edure	for St	onnin	n Slav	ve Tra	nemie	sion				
_			-							-							
Remar					-	-		-									stopped. To
		ransm		-	,			- 3	(								



No.25: Add th	ne SSIp pin in the figure. Also correct the output of the SOp pin.
Page: P.938	
Incorrect:	
	Figure 15-104. Timing Chart of Slave Transmission (in Single-Transmission Mode) (Type 1: DAPmn = 0, CKPmn = 0)
	(1)pe 1. DArmin = 0, CArmin = 0)
_	
SSmn	<u> </u>
STmn	
SEmn	-
SDRmn	Transmit data 1 Transmit data 2 Transmit data 3
SCKp pin	
SOp pin	Transmit data 1 Transmit data 2 Transmit data 3
Shift register mn	XXXX Shift operation XXX Shift operation XXXX Shift operation XXX
INTCSIp	
	Data transmission (8-bit length)
TSFmn	
Correct: For slav	ve transmission, add "SSIp pin signal" to the figure.
	Figure 15-104. Timing Chart of Slave Transmission (in Single-Transmission Mode)
	(Type 1: DAPmn = 0, CKPmn = 0)
SSmn _	
STmn _	ſ
SEmn _	
SDRmn -	Transmit data 1 X Transmit data 2 X Transmit data 3
 SCKp pin	
SOp pin	Hi-Z Transmit data 1 Transmit data 2 Transmit data 3 Transmit data 3
Shift register	
mn _	XX_X_Shift operation_XXXX_Shift operation_XXXX_Shift operation_XX
INTCSIp _	Data transmission (8-bit length)
TSFmn	
_	
SSIp pin	





г

age: P.944 correct:																
	Figure	e 15-10	08. E	Examp		Conte SI00,		-				ceptic	on of S	SPI Fi	Inctio	n
(e) Seria	al outp	ut ena	able re	egiste	r m (S	OEm)	The	e regis	ster th	at not	used	in thi	s moc	le.		
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 ×	SOEm0 ×
(f)Seria	il chan	inel st	art re	gister	.m.(SS	Sm)	Sets	only t	he bit	s of th	ie tarç	get ch	annel	to 1.		
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1
(g) Seria	al slave	e sele	ct ena	ble re	gister	: m (S	SEm)								nd SSI e mode	-
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSEm1 0/1	SSEm0 0/1
prrect:	Figure	1E 4	00 1			Conto	nto of	Bagir	toro f				n of f		motio	-
orrect: (e) Seria	Figure			-	(C	SI00,	CSI01	, CSI1	0, CS	111) (2	/2)	-			inctio	n
	-			-	(C	SI00,	CSI01	, CSI1	0, CS	111) (2	/2)	-			Inctio	n 0
	al outp	ut ena	able re	egiste	(C r m (S	:SI00, OEm)	CSI01 The	, CSI1 e regis	0, CS ster th	l11) (2 at not	/2) used	in thi	s mod	le.	1	0
<b>(e) Seria</b> SOEm	al outp	<b>ut ena</b> 14 0	able re 13 0	egiste 12 0	(C r m (S 11 0	<b>OEm)</b> 10	CSI01 The 9 0	, CSI1 e regis 8 0	0, CS ster th 7 0	111) (2 at not 6 0 s the 3	/2) used 5 0 SSI00	in this 4 0 , SSI0	s moc 3 0 1, SSI	<b>le.</b> 2 0	1 SOEm1	0 SOEm0 × 11 pin
<b>(e) Seria</b> SOEm	al outp 15 0	<b>ut ena</b> 14 0	able re 13 0	egiste 12 0	(C r m (S 11 0	<b>OEm)</b> 10	CSI01 The 9 0	, CSI1 e regis 8 0	0, CS ster th 7 0	111) (2 at not 6 0 s the 3	/2) used 5 0 SSI00	in this 4 0 , SSI0	s moc 3 0 1, SSI	<b>le.</b> 2 0	1 SOEm1 ×	0 SOEm0 × 11 pin
<b>(e) Seria</b> SOEm	al outp 15 0	ut ena 14 0 e selec	able re 13 0 ct ena	egister 12 0 ble re	(C r m (S 11 0 gister	SI00, OEm) 10 0	CSI01 The 9 0 SEm)	, CSI1 e regis 8 0 Ca in	0, CS ster th 7 0 ontrol puts c	at not 6 0 s the 3 of the	/2) used 5 0 SSI00 target	in this 4 0 , SSI0 s chan	s moc 3 0 1, SSI nel in	de. 2 0 10, ar slave	1 SOEm1 ×	0 SOEm0 × 11 pin e. 0
(e) Seria SOEm (f) Seria	al outp 15 0 al slave 15 0	ut ena 14 0 e selec 14 0	able re 13 0 ct ena 13 0	egister 12 0 ble re 12 0	(C r m (S 11 0 gister 11	<b>OEm)</b> 10 0 <b>m (S</b> 10 0	CSI01 The 9 0 SEm) 9 0	, CSI1 e regis 8 0 C4 in 8 0	0, CS ster th 7 0 ontrol puts o 7 0	at not 6 0 s the 6 0 f the 6	/2) used 5 0 SSI00 target 5 0	in this 4 0 , SSI0 chan 4 0	s mod 3 0 1, SSI nel in 3 0	<b>10, ar</b> <b>10, ar</b> <b>slave</b> 2 0	1 SOEm1 × and SSI e mode 1 SSEm1	0 SOEm0 × 11 pin 2. 0 SSEm0
(e) Seria SOEm (f) Seria SSEm	al outp 15 0 al slave 15 0	ut ena 14 0 e selec 14 0	able re 13 0 ct ena 13 0	egister 12 0 ble re 12 0	(C r m (S 11 0 gister 11	<b>OEm)</b> 10 0 <b>m (S</b> 10 0	CSI01 The 9 0 SEm) 9 0	, CSI1 e regis 8 0 C4 in 8 0	0, CS ster th 7 0 ontrol puts o 7 0	at not 6 0 s the 6 0	/2) used 5 0 SSI00 target 5 0	in this 4 0 , SSI0 chan 4 0	s mod 3 0 1, SSI nel in 3 0	<b>10, ar</b> <b>10, ar</b> <b>slave</b> 2 0	1 SOEm1 × and SSI e mode 1 SSEm1	0 SOEm0 × 11 pin 2. 0 SSEm0
SOEm <b>(f) Seria</b> SSEm	al outp 15 0 al slave 15 0	ut ena 14 0 e selec 14 0	able re 13 0 ct ena 13 0	egister	(C r m (S 11 0 gister 11 0 m (SS	SIOO, OEm) 10 0 m (S: 10 0 Sm)	CSI01 The 9 0 SEm) 9 0 Sets	, CSI1 e regis 8 0 Ca in 8 0 only t	0, CS ster th 7 0 ontrol puts o 7 0 he bit	at not 6 0 s the 6 0 s the 6 0 s of the	/2) used 5 0 SSI00 target 5 0	in this 4 0 , SSI0 chan 4 0	s moc 3 0 1, SSI nel in 3 0	de. 2 0 10, ar slave 2 0 to 1.	1 SOEm1 × and SSI e mode 1 SSEm1 0/1	0 SOEm0 × 11 pin 2. 0 SSEm0 0/1
(e) Seria SOEm (f) Seria SSEm (g) Seria	al outp 15 0 al slave 15 0 al chan 15	ut ena 14 0 e selec 14 0 nel st 14	able re 13 0 ct ena 13 0 cart re 13	egister 12 0 ble re 12 0 gister 12	(C r m (S 11 0 gister 11 0 m (SS 11	SI00, OEm) 10 0 m (S: 10 0 Sm) 10	CSI01 The 9 0 SEm) 9 0 Sets 9	, CSI1 = regis 8 0 Ca in 8 0 only t 8	0, CS ster th 7 0 ontrol puts c 7 0 he bit 7	111) (2 at not 6 0 s the 6 0 s of the 6	/2) used 5 0 SSI00 target 5 0 ne targ 5	in this 4 0 , SSI0 c han 4 0 get ch 4	s mod 3 0 1, SSI nel in 3 0 annel 3	de. 2 0 10, ar slave 2 0 to 1. 2	1 SOEm1 × mode 1 SSEm1 0/1	0 SOEm0 × 11 pin >. 0 SSEm0 0/1 0 SSEm0 0/1
(e) Seria SOEm (f) Seria SSEm (g) Seria	al outp 15 0 al slave 15 0 al chan 15	ut ena 14 0 e selec 14 0 nel st 14	able re 13 0 ct ena 13 0 cart re 13	egister 12 0 ble re 12 0 gister 12	(C r m (S 11 0 gister 11 0 m (SS 11	SI00, OEm) 10 0 m (S: 10 0 Sm) 10	CSI01 The 9 0 SEm) 9 0 Sets 9	, CSI1 = regis 8 0 Ca in 8 0 only t 8	0, CS ster th 7 0 ontrol puts c 7 0 he bit 7	111) (2 at not 6 0 s the 6 0 s of the 6	/2) used 5 0 SSI00 target 5 0 ne targ 5	in this 4 0 , SSI0 c chan 4 0 get cha 4	s mod 3 0 1, SSI nel in 3 0 annel 3	de. 2 0 10, ar slave 2 0 to 1. 2	1 SOEm1 × mod SSI e mode 1 SSEm1 0/1	0 SOEm0 × 11 pin >. 0 SSEm0 0/1 0 SSEm0



ge: P.947 correct: SSmn STmn	Figure 15-112. T				
	Figure 15-112. T				
			f Slave Reception (in Singl	e-Reception Mode)	
		(Type 1: [	0APmn = 0, CKPmn = 0)		
-					
SEmn				Dessive data 2	
SDRmn		χ	Receive data 1	Receive data 3 Receive data 2	
			ead	▲ Read Rea	ad 🛦
SCKp pin			uuuuu		
SIp pin	Receive data	1	Receive data 2	Receive data 3	
Shift —— egister mn ——	Reception & shift o	peration	Reception & shift operation	Reception & shift operation	
		i	r		L
	Data reception (8-b	it length)	Data reception (8-bit length)	Data reception (8-bit length)	
TSFmn					
rrect: For slave	reception, add "SSIp	pin signal" to th	e figure.		
	Figure 15-112. T	iming Chart o	f Slave Reception (in Singl	e-Reception Mode)	
		(Туре 1: Г	0APmn = 0, CKPmn = 0)		
SSmn 🔽	1				
STmn					]
SEmn					
SDRmn		γ	Receive data 1	Receive data 3 -	7
			Read	▲Read Read	
SCKp pin		╷╷┍╷╷╶╸			
SIp pin		data 1	Receive data 2	Receive data 3	
Shift register	Reception & shif	t operation	Reception & shift operation	Reception & shift operation X	
INTCSIp					
·	Data reception (8-	bit length)	Data reception (8-bit length)	Data reception (8-bit length)	
TSFmn					
SSIp pin					
F	<u> </u>				

RENESAS

г

	• •	Tegi			5 0.0	• …				ne sai		uy us			guio	
Page: P.951																
ncorrect: Figure	15-11	1 Ev	amnl	a of C	onton	ts of F	Ponist	ore fo	r Slav	o Tran	emiee	sion/R	ocont	ion of		unction
igure	13-11-	τ. LA	ampi				-			l11) (2		51011/13	ecept			unction
(e) Seria	al outp	ut ena	able re	egiste	r m (S	OEm)	Se	ts onl	y the	bits of	the ta	arget	chann	el to	1.	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	SOEm0 0/1
(f)Seria	al char	nel st	art re	gister	.m.(SS	Sm)	Sets	only t	he bit	s of th	ne targ	get ch	annel	to 1.		
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1
(g) Seria	al slave	e sele	ct ena	ıble re	gister	. m (S	SEm)	C	ontrol	s the	SS100,	, SSI0	1, SSI	10, ar	nd SSI	11 pin
	45		40	40		10	0		-		-				e mode	
SSEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSEII	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1	SSEm0 0/1
orrect:																
Figure	15-114	1 Fy						-		_						
		τ. <b>Ε</b> Λ	ampie				-			e Tran I11) (2		sion/R	ecept	ion of	f SPI F	unction
(e) Seria	al outp			egiste	(C	SI00,	CSI01	, CSI1	0, CS	111) (2	/2)			el to		unction
(e) Seria	al outp 15				(C	SI00,	CSI01	, CSI1	0, CS	111) (2	/2)					o
<b>(e) Seria</b> SOEm	-	ut ena	able re	egiste	(C r m (S	SI00, OEm)	CSI01 Se	, CSI1 ts onl	0, CS	l11) (2 bits of	/2) the ta	arget	chann	el to	<b>1</b> .	
	15 0	<b>ut ena</b> 14 0	able re 13 0	12 0	(C r m (S 11 0	<b>OEm)</b> 10	CSI01 Se 9 0	, CSI1 ts onl 8 0	v the 7 0 ontrol	l11) (2 bits of 6 0 s the	/2) f the ta 5 0 SSI00;	arget o 4 0 , SSI0	chann 3 0 1, SSI	el to 2 0	1. 1 SOEm1 0/1	0 SOEm0 0/1 11 pin
SOEm	15 0 al slave	ut ena 14 0 e selec	able re 13 0 ct ena	12 0 ble re	(C r m (S 11 0	SI00, OEm) 10 0	CSI01 Se 9 0 SEm)	, CSI1 ts onl 8 0 C in	ontrol	l11) (2 bits of 6 0 s the of the	/2) f the ta 5 0 SSI00, target	arget o 4 0 , SSI0 : chan	chann 3 0 1, SSI nel in	el to 2 0 10, ar slave	1. SOEm1 0/1	0 SOEm0 0/1 11 pin
SOEm	15 0	<b>ut ena</b> 14 0	able re 13 0	12 0	(C r m (S 11 0	SI00, OEm) 10 0	CSI01 Se 9 0	, CSI1 ts onl 8 0	0, CS y the 7 0 ontrol	l11) (2 bits of 6 0 s the	/2) f the ta 5 0 SSI00;	arget o 4 0 , SSI0	chann 3 0 1, SSI	el to 2 0	1. 1 SOEm1 0/1 nd SSI a mode 1	0 SOEm0 0/1 11 pin 2. 0
SOEm	15 0 al slave	ut ena 14 0 e selec	able re 13 0 ct ena	12 0 ble re	(C r m (S 11 0	SI00, OEm) 10 0 • m (S	CSI01 Se 9 0 SEm)	, CSI1 ts onl 8 0 C in	ontrol	l11) (2 bits of 6 0 s the of the	/2) f the ta 5 0 SSI00, target	arget o 4 0 , SSI0 : chan	chann 3 0 1, SSI nel in	el to 2 0 10, ar slave	1. 1 SOEm1 0/1 nd SSI a mode 1	0 SOEm0 0/1 11 pin
SOEm (f) Seria SSEm	15 0 al slave 15 0	ut ena 14 0 e selec 14 0	able re 13 0 ct ena 13 0	egiste 12 0 ble re 12 0	(C r m (S 11 0 •gister 11 0	SI00, OEm) 10 0 m (S: 10 0	CSI01 Se 9 0 SEm) 9 0	, CSI1 ts onl 8 0 C in 8 0	ontrol puts o 7 0	l11) (2 bits of 6 0 s the 6 6 0	/2) the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSI0 chan 4 0	chann 3 0 1, SSI nel in 3 0	el to 2 0 10, ar slave 2 0	1. SOEm1 0/1 nd SSI e mode 1 SSEm1	0 SOEm0 0/1 11 pin 2. 0 SSEm0
SOEm (f) Seria	15 0 al slave 15 0	ut ena 14 0 e selec 14 0	able re 13 0 ct ena 13 0	egiste 12 0 ble re 12 0	(C r m (S 11 0 •gister 11 0	SI00, OEm) 10 0 m (S: 10 0	CSI01 Se 9 0 SEm) 9 0 Sets	, CSI1 ts onl 8 0 C in 8 0 only t	ontrol puts o 7 0	IIII) (2 bits of 6 0 s the of the 6 0 s of th	/2) f the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSIO chan 4 0 get cha	chann 3 0 1, SSI nel in 3 0	el to 2 0 10, ar slave 2 0	1. SOEm1 0/1 nd SSI e mode 1 SSEm1	0 SOEm0 0/1 11 pin 2. 0 SSEm0
SOEm (f) Seria SSEm	15 0 15 0 al char	ut ena 14 0 e selec 14 0	able re 13 0 ct ena 13 0	egiste 12 0 ble re 12 0 gister	(C r m (S 11 0 egister 11 0 m (SS	SIOO, OEm) 10 0 m (S: 10 0 Sm)	CSI01 Se 9 0 SEm) 9 0	, CSI1 ts onl 8 0 C in 8 0	ontrol puts o 7 0	l11) (2 bits of 6 0 s the 6 6 0	/2) the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSI0 chan 4 0	chann 3 0 1, SSI nel in 3 0 annel	el to 2 0 10, ar slave 2 0 to 1.	1. 1 SOEm1 0/1 nd SSI e mode 1 SSEm1 0/1	0 SOEm0 0/1 11 pin 2. 0 SSEm0 0/1
SOEm (f) Seria SSEm (g) Seria	15 0 15 0 15 0	ut ena 14 0 e selec 14 0 nnel st 14	able re 13 0 ct ena 13 0 ct art re 13	egiste 12 0 ble re 12 0 gister 12	(C r m (S 11 0 egister 11 0 m (SS 11	SI00, OEm) 10 0 m (S: 10 0 Sm) 10	CSI01 Se 9 0 SEm) 9 0 Sets 9	, CSI1 ts onl 8 0 C in 8 0 0 only t 8	ontrol puts o 7 0 0 0 0 0	III) (2 bits of 6 0 s the 6 0 s of the 6 0	/2) f the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSIO chan 4 0 get cha 4	chann 3 0 1, SSI nel in 3 0 annel 3	el to 2 0 10, ar slave 2 0 to 1. 2	1. 1 SOEm1 0/1 nd SSI e mode 1 SSEm1 0/1 1 SSm1	0 SOEm0 0/1 11 pin 2. 0 SSEm0 0/1 0/1
SOEm (f) Seria SSEm (g) Seria	15 0 15 0 15 0	ut ena 14 0 e selec 14 0 nnel st 14	able re 13 0 ct ena 13 0 ct art re 13	egiste 12 0 ble re 12 0 gister 12	(C r m (S 11 0 egister 11 0 m (SS 11	SI00, OEm) 10 0 m (S: 10 0 Sm) 10	CSI01 Se 9 0 SEm) 9 0 Sets 9	, CSI1 ts onl 8 0 C in 8 0 0 only t 8	ontrol puts o 7 0 0 0 0 0	III) (2 bits of 6 0 s the 6 0 s of the 6 0	/2) f the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSIO chan 4 0 get cha 4	chann 3 0 1, SSI nel in 3 0 annel 3	el to 2 0 10, ar slave 2 0 to 1. 2	1. 1 SOEm1 0/1 nd SSI e mode 1 SSEm1 0/1 1 SSm1	0 SOEm0 0/1 11 pin 2. 0 SSEm0 0/1 0/1
SOEm (f) Seria SSEm (g) Seria	15 0 15 0 15 0	ut ena 14 0 e selec 14 0 nnel st 14	able re 13 0 ct ena 13 0 ct art re 13	egiste 12 0 ble re 12 0 gister 12	(C r m (S 11 0 egister 11 0 m (SS 11	SI00, OEm) 10 0 m (S: 10 0 Sm) 10	CSI01 Se 9 0 SEm) 9 0 Sets 9	, CSI1 ts onl 8 0 C in 8 0 0 only t 8	ontrol puts o 7 0 0 0 0 0	III) (2 bits of 6 0 s the 6 0 s of the 6 0	/2) f the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSIO chan 4 0 get cha 4	chann 3 0 1, SSI nel in 3 0 annel 3	el to 2 0 10, ar slave 2 0 to 1. 2	1. 1 SOEm1 0/1 nd SSI e mode 1 SSEm1 0/1 1 SSm1	0 SOEm0 0/1 11 pin 2. 0 SSEm0 0/1 0/1
SOEm (f) Seria SSEm (g) Seria	15 0 15 0 15 0	ut ena 14 0 e selec 14 0 nnel st 14	able re 13 0 ct ena 13 0 ct art re 13	egiste 12 0 ble re 12 0 gister 12	(C r m (S 11 0 egister 11 0 m (SS 11	SI00, OEm) 10 0 m (S: 10 0 Sm) 10	CSI01 Se 9 0 SEm) 9 0 Sets 9	, CSI1 ts onl 8 0 C in 8 0 0 only t 8	ontrol puts o 7 0 0 0 0 0	III) (2 bits of 6 0 s the 6 0 s of the 6 0	/2) f the ta 5 0 SSI00, target 5 0	arget o 4 0 , SSIO chan 4 0 get cha 4	chann 3 0 1, SSI nel in 3 0 annel 3	el to 2 0 10, ar slave 2 0 to 1. 2	1. 1 SOEm1 0/1 nd SSI e mode 1 SSEm1 0/1 1 SSm1	0 SOEm0 0/1 11 pin 2. 0 SSEm0 0/1 0/1

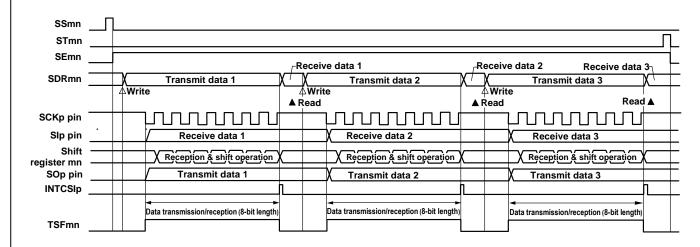


#### No.30: Add the SSIp pin in the figure. Also correct the output of the SOp pin.



Incorrect:

Figure 15-118. Timing Chart of Slave Transmission/Reception (in Single-Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)

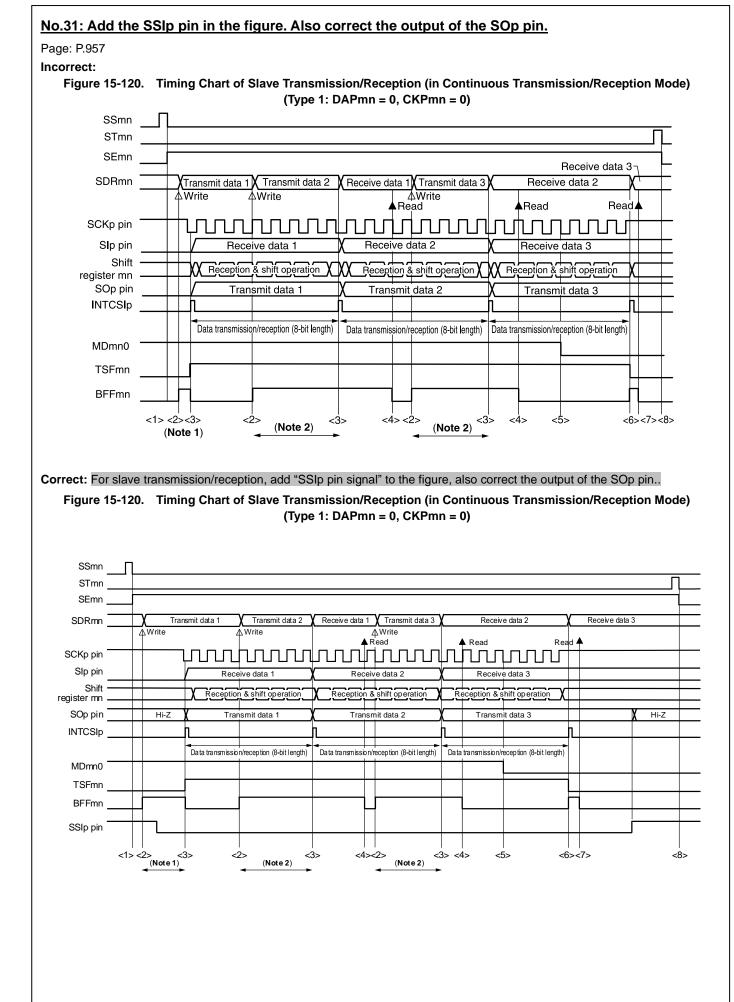


Correct: For slave transmission/reception, add "SSIp pin signal" to the figure, also correct the output of the SOp pin..

Figure 15-118. Timing Chart of Slave Transmission/Reception (in Single-Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)

SSmn									
STmn									
SEmn _				<sub>Γ</sub> R	Receive data 1	_ Re	ceive data 2	$_{\Gamma}$ Receive data	3
SDRmn		Transmit data 1			Transmit data 2		Transmit data 3	χ /	
	4	∆ Write			∆Write ead	∆' ▲ Rea	Write ad	▲ Read	
SCKp pin			M				linnin		
SIp pin			Receive data 1		Receive data 2		Receive data 3		
- Shift - register mn			Reception & shift operation		X Reception & shift operation X		X Reception & shift operation X		
SOp pin	Hi-Z	X	Transmit data 1		Transmit data 2		Transmit data 3		Hi-Z
INTCSIp				Ц	[ſ	∟		r	
TSFmn _			Da ta transmission/reœption (8-bit length)		Da ta transmission/re ce ption (8-bit length)		▲ Da ta transmission/reœption (8-bit length)		
SSIp pin									







## No.32: Add "Reception end interrupt" to [Interrupt function].

Page: P.962

## Incorrect:

### 15.7 Operation of UART (UART0, UART1) Communication

[Interrupt function]

• Transfer end interrupt/buffer empty interrupt

------

### Correct:

## 15.7 Operation of UART (UART0, UART1) Communication

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt
- Reception end interrupt

## No.33: Correct reference figure number.

Page: P.1101

Incorrect:

### 16.6.2 Example Timing Charts of Slave to Master Communications

The meanings of <1> to <7> in Figure 18-33 (1) Start condition ~ address ~ data are explained below.

Correct:

#### 16.6.2 Example Timing Charts of Slave to Master Communications

The meanings of <1> to <7> in Figure 16-33 (1) Start condition ~ address ~ data are explained below.

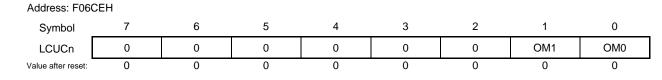
## No.34: Add "Caution" regarding LIN reset mode transitions.

Page: P.1129

Correct:

#### 17.2.1 LIN Registers for Master Mode

#### (16) LIN/UART Control Register (LCUCn)



Caution When transitioning from LIN mode with timeout error detection enabled (FTERE bit of LEDEn register is set to 1) to LIN reset mode and then to LIN mode, clear the error flag according to the procedure described in "17.3.1 LIN Reset Mode".

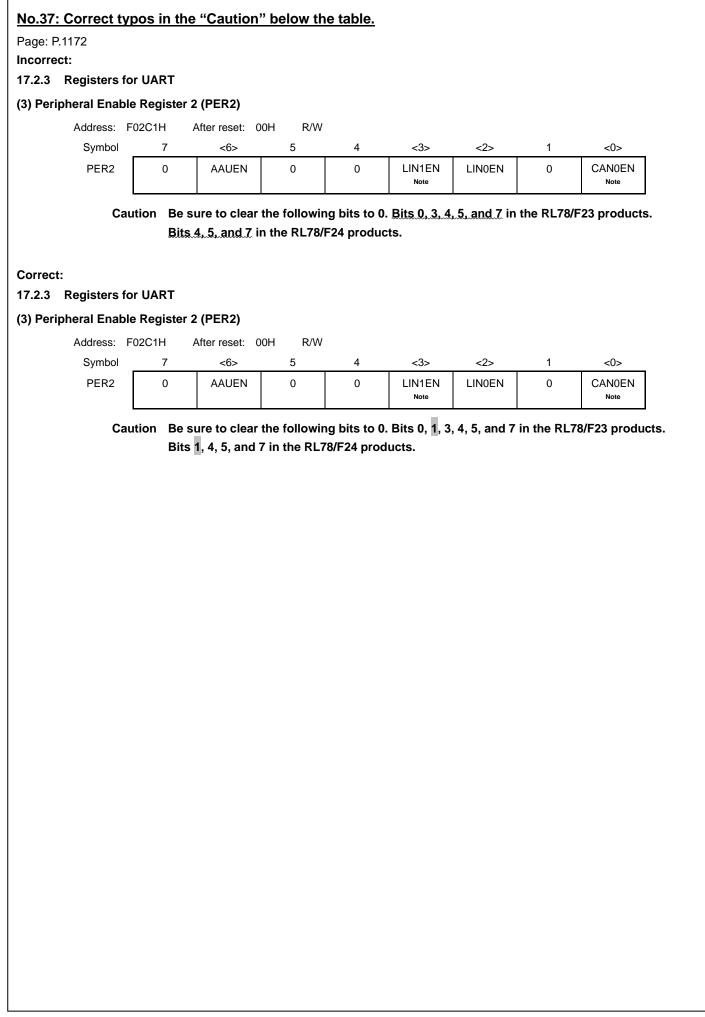


Г

	pos in th	e "Cautior	n" below th	e table.				
Page: P.1143								
Incorrect:								
17.2.2 LIN Register	rs for Slav	e Mode						
(3) Peripheral Enable	e Register	2 (PER2)						
Address: F	02C1H	After reset:	00H R/W					
Symbol	7	<6>	5	4	<3>	<2>	1	<0>
PER2	0	AAUEN	0	0	LIN1EN Note	LINOEN	0	CAN0EN Note
Cau			r the followir in the RL78/			.5. and 7 in 1	the RL78/F	
Correct: 17.2.2 LIN Register	rs for Slav	e Mode						
(3) Peripheral Enable	e Register	2 (PER2)						
Address: F	02C1H	After reset:	00H R/W					
Symbol	7	<6>	5	4	<3>	<2>	1	<0>
PER2	0	AAUEN	0	0	LIN1EN Note	LINOEN	0	CAN0EN Note
17.2.2 LIN Register (15) LIN/UART Contr Address: F06CEH Symbol	rol Registe		5	4	3	2	1	0
LCUCn	0	0	0	0	0	0	OM1	OM0
Value after reset:	0	0	0	0	0	0	0	0
to 1) to Lll "17.3.1 L			n to LIN mo	de, clear th				EDEn register is se
		<i>l</i> ode".			e error flag	according t	o the proc	edure described



## RENESAS TECHNICAL UPDATE TN-RL\*-A0139A/E



# No.38: Add the "Caution" about transitioning to LIN reset mode when timeout error detection is enabled. Page: P.1202 Correct: 17.3.1 LIN Reset Mode Caution When transitioning from LIN mode with timeout error detection enabled (FTERE bit or TERE bit of LEDEn register is set to 1) to LIN reset mode and then to LIN mode, clear the error flags using the procedure shown in the figure below. If LINnEN bit of PER2 register is set to 0 to transition to LIN mode, it is not necessary to clear the error flag. Start LINnSTAMK/LINnMK ← 1 Mask the LIN error interrupt. LCUCn.OM0 ← 0 Transition to LIN reset mode. No LMSTn.OMM0 = 0? Check for transition to LIN reset mode. Yes (User reset process) LCUCn.OM0 ← 1 LIN reset mode release request. No LMSTn.OMM0 = 1? Check for LIN reset mode release. Yes LESTn ← 0x00 Clear the error flag. LINnSTAIF/LINnIF ← 0 Clear the LIN error interrupt request flag. LINnSTAMK/LINnMK ← 0 Clear the LIN error interrupt mask flag (if necessary). End



### No.39: Correct the typo in the DBRP[7:0] bit setting value.

## Page: P.1295

## Incorrect:

## 18.3.5 Channel 0 Data Bitrate Configuration Register (C0DCFGH, C0DCFGL)

Symbol	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
C0DCFGH	-	-	-	-		DSJV	V[3:0]		-	-	-	-		DTSE	G2[3:0]	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Symbol	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
CODCFGL	-	-	-		DT	SEG1[4	:0]					DBRI	P[7:0]			
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

#### • DBRP[7:0]

When these bits are set to P (0 to 1023), the baud rate prescaler divides fcan by P + 1.

The CAN0 Tq clock (fcantoo) is obtained by the CAN communication clock (fcan) and setting the clock division ratio with the DBRP[9:0] bits and one clock cycle of the CAN0 Tq clock is 1 Time Quantum (Tq).

These bits cannot be written in CH\_OPERATION or CH\_SLEEP mode.

Write these bits only when the RS-CANFD lite channel is in CH\_RESET or CH\_HALT mode.

#### Correct:

## 18.3.5 Channel 0 Data Bitrate Configuration Register (C0DCFGH, C0DCFGL)

Symbol	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
C0DCFGH	-	-	Ι	١		DSJV	V[3:0]		Ι	I	-	Ι		DTSE	G2[3:0]	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Symbol	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
CODCFGL	-	-	-		DT	SEG1[4	:0]					DBR	P[7:0]			
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

## • DBRP[7:0]

When these bits are set to P (0 to 255), the baud rate prescaler divides  $f_{CAN}$  by P + 1.

The CAN0 Tq clock (f<sub>CANTQ0</sub>) is obtained by the CAN communication clock (f<sub>CAN</sub>) and setting the clock division ratio with the DBRP[9:0] bits and one clock cycle of the CAN0 Tq clock is 1 Time Quantum (Tq).

These bits cannot be written in CH\_OPERATION or CH\_SLEEP mode.

Write these bits only when the RS-CANFD lite channel is in CH\_RESET or CH\_HALT mode.



## No.40: Add description of high-speed DTC to the specification explanation for "Repeat mode" in

## "Transfer mode".

Page: P.1500

Incorrect:

	Table 19-1. DTC Specifications							
Item		Specification						
Transfer mode	Normal mode	Transfers end on completion of the transfer causing the DTCCTj and HDTCCTm registers value to change from 1 to 0.						
	Repeat mode	On completion of the transfer causing the values of the DTCCTj and HDTCCTm registers to change from 1 to 0, the repeat area address is initialized and the DTRLDj register value is reloaded to the DTCCTj and HDTCCTm registers to continue transfers.						

#### Correct:

Table 19-1.	<b>DTC Specifications</b>
	Die opecifications

lte	m	Specification
Transfer mode	Normal mode	Transfers end on completion of the transfer causing the DTCCTj and HDTCCTm registers value to change from 1 to 0.
	Repeat mode	On completion of the transfer causing the values of the DTCCTj or HDTCCTm register to change from 1 to 0, the repeat area address is initialized, the value of DTRLDj is reloaded to the DTCCTj register, or the value of HDTRLDm is reloaded to the HDTCCTm register, and the DTC continues the transfer.

## No.41: Add Note regarding DTCCR23 register.

Page: P.1529

Incorrect:

#### 19.3.4 Chain Transfers

When the CHNE bit in the DTCCRj (j = 0. to .22) register is 1 (chain transfers enabled), multiple data transfers can be continuously performed by one activation source.

#### Correct:

### 19.3.4 Chain Transfers

When the CHNE bit in the DTCCRj (j = 0 to  $22^{Note}$ ) register is 1 (chain transfers enabled), multiple data transfers can be continuously performed by one activation source.

#### Note Set the CHNE bit in the DTCCR23 register to 0 (chain transfers disabled).

## No.42: Correct typo in general-purpose register addresses.

Page: P.1531

Incorrect:

## 19.4.2 Allocation of DTC Control Data Area and DTC Vector Table Area

The areas where the DTC control data and vector table can be allocated differ, depending on the usage conditions.

• It is prohibited to use the general-purpose register (**FFF00H to FFEE0H**) space as the DTC control data area or DTC vector table area.

## Correct:

## 19.4.2 Allocation of DTC Control Data Area and DTC Vector Table Area

The areas where the DTC control data and vector table can be allocated differ, depending on the usage conditions.

• It is prohibited to use the general-purpose register (FFEE0H to FFEFFH) space as the DTC control data area or DTC vector table area.



Address: FF Symbol IF2H	FFD1H A 7		· • · · · · · • · • ·	t Request i	-lag Regist	ers (IFxL, IF	xH) (2/2)	
-	7	fter reset: 00	OH R/W					
IF2H		<6>	5	<4>	<3>	<2>	<1>	<0>
	0	TMIF13	TMIF12	TMIF11	TMIF10	RCANGERRIF	RCANGREFRIF	RCAN0TRMIF
r <b>ect:</b> Address: FF Symbol	-		t <b>of Interrup</b> DH R/W <5>	ot Request I	Flag Regist	ers (IFxL, IF	x <b>H) (2/2)</b> <1>	<0>
IF2H	0	TMIF13	TMIF12	TMIF11	TMIF10	RCANGERRIF		CAN0TRMIF
prrect: Fiq Address: F0	-		-	ource Deter	mination Fl	ag Register	0 (INTFLG0	)
Address. FL								
		iter reset: 00 6		4	3	2	1	0
Symbol INTFLG0 <b>Note</b>	7 INTFLG07 Notes 3, 5 4. Even i the DT	6 INTFLG06 <sub>Notes 2, 4</sub> f the RPTIN <sup>*</sup> C module is	5 0 T bit in the I in repeat m	ode), when	the compara	2 INTFLG02 Notes 1, 5 o 23) is set t ator detection	0 interrupt	source is ge
Symbol INTFLGO Note	7 INTFLG07 Notes 3, 5 4. Even i the DT the IN detect	6 INTFLG06 Notes 2, 4 f the RPTIN TC module is TFLG06 bit i ion 0 interro	5 0 T bit in the I in repeat m s set to 1. <b>F</b> upt) in Figu	0 DTCCRj reg ode), when For details, re 21-1. Bas	0 ister (j = 0 t the compara see (A) Inte sic Configu	INTFLG02 Notes 1, 5 o 23) is set t ator detection ernal maska ration of Int	INTFLG01 Note 5 0 0 (disablin 0 interrupt 5 ble interrup errupt Func	INTFLG00 Note 5 g the interru source is ge t (only con tion.
Symbol INTFLGO Note	7 INTFLG07 Notes 3, 5 4. Even i the DT the IN detect	6 INTFLG06 Notes 2, 4 f the RPTIN C module is TFLG06 bit i ion 0 interro	5 0 T bit in the I in repeat m s set to 1. F upt) in Figu	0 DTCCRj reg ode), when For details, re 21-1. Bas	0 ister (j = 0 t the compara see (A) Inte sic Configu	INTFLG02 Notes 1, 5 o 23) is set t ator detection	INTFLG01 Note 5 0 0 (disablin 0 interrupt 5 ble interrup errupt Func	INTFLG00 Note 5 g the interru source is ge t (only con tion.
Symbol INTFLGO Note	7 INTFLG07 Notes 3, 5 4. Even i the DT the IN detect	6 INTFLG06 Notes 2, 4 f the RPTIN TC module is TFLG06 bit i ion 0 interro	5 0 T bit in the I in repeat m s set to 1. F upt) in Figu	0 DTCCRj reg ode), when For details, re 21-1. Bas	0 ister (j = 0 t the compara see (A) Inte sic Configu	INTFLG02 Notes 1, 5 o 23) is set t ator detection ernal maska ration of Int	INTFLG01 Note 5 0 0 (disablin 0 interrupt 5 ble interrup errupt Func	INTFLG00 Note 5 g the interru source is ge t (only con tion.



Γ

	t:	-		Farmer of I							
		F	igure 21-9.	Format of I	nput Swite	ch Control R	egister (ISC)	)			
	Address: F		After reset: 00								
	Symbol	7	6	5	4	≲3≳	<u>≲2≳</u>	1	≲ <b>0</b> ≳		
	ISC	0	0	0	0	ISC3	ISC2	0	ISC0		
rect:											
		F	igure 21-9.	Format of I	nput Swite	ch Control R	egister (ISC)	)			
	Address: F	0073H A	After reset: 00	H R/W							
	Symbol	7	6	5	4	3	2	1	0		
	ISC	0	0	0	0	ISC3	ISC2	0	ISC0		
	Example 1)	upt request Interrupt re	hold instructio	eld pending.				lso continu	le.		
		-	pt instruction			t hold instru t hold instru					
			equest is acce pt instruction		ed. Interrupt request hold instruction						
		able interru	pt instruction	Interru	upt reques	t hold instru	iction				



## No.47: Correct the bit symbol in the Note 2.

## Page: P.1620

## Incorrect:

Table 24-4. Reset Function Control Registers

Address	Register Name	Symbol	After Reset	Access Size
FFFA8H	Reset control flag register	RESF	00H <sup>Note 1</sup>	8
F02C9H	POR/CLM reset confirmation register	POCRES	00H Note 2	1, 8

**Notes 1.** The value after reset varies depending on the reset source.

2. The value immediately before a reset is retained when a reset is from any source other than the POR circuit.

#### Correct:

#### Table 24-4. Reset Function Control Registers

Address	Register Name	Symbol	After Reset	Access Size
FFFA8H	Reset control flag register	RESF	00H Note 1	8
F02C9H	POR/CLM reset confirmation register	POCRES	00H <sup>Note 2</sup>	1, 8

**Notes 1.** The value after reset varies depending on the reset source.

2. If the reset is from a source other than the POR circuit, the value of the POCRES0 bit immediately before the reset is retained.

## No.48: Add a description of the RESF register when an internal reset occurs to the Remark.

### Page: P.1623

Incorrect:

## 25.1 Functions of Power-on-reset Circuit

**Remark** This product incorporates multiple hardware functions that generate an internal reset signal. A flag that indicates the reset source is located in the RESF and POCRES registers for when an internal reset signal is generated by the power-on reset (POR), watchdog timer (WDT), clock monitor, voltage detector (LVD), illegal instruction execution, or illegal-memory access. The CLKRF bit of the POCRES register is set to 1 when an internal reset signal is generated by the clock monitor. The POCRES0 bit of the POCRES register is cleared to 0 by the POR reset when it has been set to 1 beforehand. For details of the POCRES and RESF registers, see **CHAPTER 24 RESET FUNCTION**.

Correct:

#### 25.1 Functions of Power-on-reset Circuit

**Remark** This product incorporates multiple hardware functions that generate an internal reset signal. A flag that indicates the reset source is located in the RESF and POCRES registers for when an internal reset signal is generated by the power-on reset (POR), watchdog timer (WDT), clock monitor, voltage detector (LVD), illegal instruction execution, or illegal-memory access.

The RESF register is not cleared (00H) and the flag is set (1) when an internal reset signal is generated due to WDT/LVD/illegal instruction execution/clock monitoring/illegal-memory access. The CLKRF bit of the POCRES register is set to 1 when an internal reset signal is generated by the clock monitor. The POCRES0 bit of the POCRES register is cleared to 0 by the POR reset when it has been set to 1 beforehand. For details of the POCRES and RESF registers, see **CHAPTER 24 RESET FUNCTION**.



## No.49: Correct the fourth bullet point description.

Page: P.1629

## Incorrect:

## 26.1 Functions of Voltage Detector

 After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in 36.4, 37.4 or 38.4 AC Characteristics. This is done by utilizing the voltage detector or controlling the externally input reset signal. Immediately after the power supply is turned off, this LSI should be placed in the STOP mode, or placed in the reset state by utilizing the voltage detector or controlling the externally input reset signal before the voltage falls below the operating range.

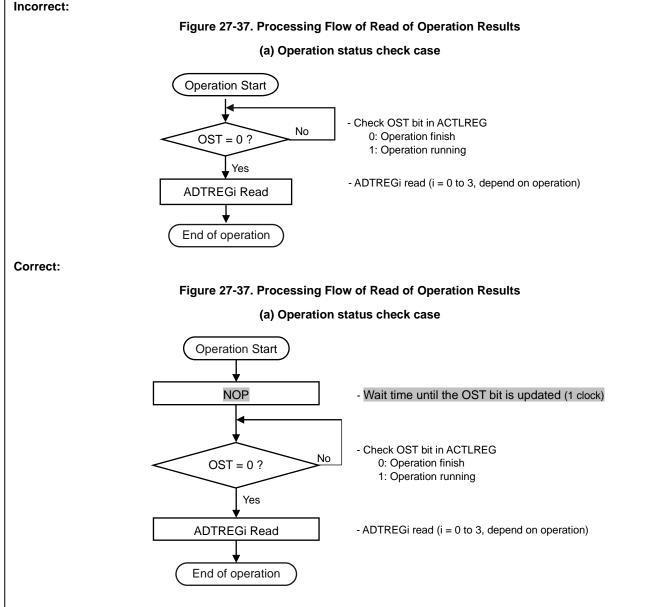
### Correct:

## 26.1 Functions of Voltage Detector

 After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in Electrical Specifications. This is done by using the voltage detector or controlling the externally input reset signal. After the power supply is turned off, this LSI should be entered in the STOP mode, or placed in the reset state by using the voltage detector or controlling the externally input reset signal before the supply voltage falls below operating range.

## No.50: Add "NOP instruction (1 cycle)" in the figure.

Page: P.1671





#### No.51: Correct typos in the internal calculation formula.

Page: P.1676

Incorrect:

27.3.2.4 Clarke and Park Transformation (Power invariant transformation)

Calculation of Clarke and Park Transformation (Power invariant transformation):

```
// AAU: Clarke & Park transformation (Power invariant transformation)
signed short sin buf, cos buf, temp16 0, temp16 1;
signed long
               temp32_0, temp32_1, temp32_2, temp32_3;
sin_buf = AAU_SIN(ADTREG2);
                                         /* -16384 to 16384: -1.0 to 1.0 << 14 */
cos buf = AAU COS(ADTREG2);
                                         /* -16384 to 16384: -1.0 to 1.0 << 14 */
temp32_0 = (signed long)ADTREG0 * 20066; /* 20066: sqrt(3/2) << 14 */</pre>
temp16 0 = (signed short)(temp32 0 >> 14U);
temp32 0 = (signed long)temp16 0 * (signed long)cos buf;
temp16_1 = ADTREG0 + ADTREG1;
temp16_1 = temp16_1 + ADTREG1;
temp32_1 = (signed long)temp16_1 * 23170; /* 23170: sqrt(2)/2 << 15 */
temp16_1 = (signed short)(temp32_1 >> 15U);
temp32_1 = (signed long)temp16_1 * (signed long)cos_buf;
temp32_1 = -temp32_1;
temp32_2 = (signed long)temp16_1 * (signed long)sin_buf;
ADTREG0 = (unsigned short)((temp32_0 - temp32_2) >> 14U);
temp32_3 = (signed long)temp16_0 * (signed long)sin_buf;
ADTREG1 = (unsigned short)((temp32_1 - temp32_3) >> 14U);
```

Correct:

#### 27.3.2.4 Clarke and Park Transformation (Power invariant transformation)

Calculation of Clarke and Park Transformation (Power invariant transformation):

```
// AAU: Clarke & Park transformation (Power invariant transformation)
signed short sin_buf, cos_buf, temp16_0, temp16_1;
               temp32_0, temp32_1, temp32_2, temp32_3;
signed long
sin_buf = AAU_SIN(ADTREG2);
                                        /* -16384 to 16384: -1.0 to 1.0 << 14 */
cos_buf = AAU_COS(ADTREG2);
                                        /* -16384 to 16384: -1.0 to 1.0 << 14 */
temp32_0 = (signed long)ADTREG0 * 20066; /* 20066: sqrt(3/2) << 14 */
temp16 0 = (signed short)(temp32 0 >> 14U);
temp32 0 = (signed long)temp16 0 * (signed long)cos buf;
temp32 1 = ADTREG0 + ADTREG1;
temp32 1 = temp32 1 + ADTREG1;
temp32_1 = temp32_1 * 23170;
                                         /* 23170: sart(2)/2 << 15 */
temp16_1 = (signed short)(temp32_1 >> 15U);
temp32_1 = (signed long)temp16_1 * (signed long)cos_buf;
temp32 \ 1 = -temp32 \ 1;
temp32_2 = (signed long)temp16_1 * (signed long)sin_buf;
ADTREG0 = (unsigned short)((temp32_0 - temp32_2) >> 14U);
temp32_3 = (signed long)temp16_0 * (signed long)sin_buf;
ADTREG1 = (unsigned short)((temp32 1 - temp32 3) >> 14U);
```



#### No.52: Correct typos in the internal calculation formula.

Page: P.1678

Incorrect:

27.3.2.5 Clarke and Park Transformation (Amplitude invariant transformation)

Calculation of Clarke and Park Transformation (Amplitude invariant transformation):

```
// AAU: Clarke & Park transformation (Amplitude invariant transformation)
signed short sin_buf, cos_buf, temp16_0, temp16_1;
signed long temp32_0, temp32_1, temp32_2, temp32_3;
sin_buf = AAU_SIN(ADTREG2);
                                        /* -16384 to 16384: -1.0 to 1.0 << 14 */
cos_buf = AAU_COS(ADTREG2);
                                       /* -16384 to 16384: -1.0 to 1.0 << 14 */
temp16_0 = ADTREG0;
temp32_0 = (signed long)temp16_0 * (signed long)cos_buf;
temp16 1 = ADTREG0 + ADTREG1;
temp16 1 = temp16 1 + ADTREG1;
temp32_1 = (signed long)temp16_1 * 18919; /* 18919: sqrt(3)/3 << 15 */</pre>
temp16 1 = (signed short)(temp32 1 >> 15U);
temp32_1 = (signed long)temp16_1 * (signed long)cos_buf;
temp32_1 = -temp32_1;
temp32_2 = (signed long)temp16_1 * (signed long)sin_buf;
ADTREG0 = (unsigned short)((temp32_0 - temp32_2) >> 14U);
temp32_3 = (signed long)temp16_0 * (signed long)sin_buf;
ADTREG1 = (unsigned short)((temp32 1 - temp32 3) >> 14U);
```

Correct:

27.3.2.5 Clarke and Park Transformation (Amplitude invariant transformation)

Calculation of Clarke and Park Transformation (Amplitude invariant transformation):

```
// AAU: Clarke & Park transformation (Amplitude invariant transformation)
signed short sin_buf, cos_buf, temp16_0, temp16_1;
signed long temp32_0, temp32_1, temp32_2, temp32_3;
sin_buf = AAU_SIN(ADTREG2);
                                       /* -16384 to 16384: -1.0 to 1.0 << 14 */
cos buf = AAU COS(ADTREG2);
                                       /* -16384 to 16384: -1.0 to 1.0 << 14 */
temp16 0 = ADTREG0;
temp32_0 = (signed long)temp16_0 * (signed long)cos_buf;
temp32 1 = ADTREG0 + ADTREG1;
temp32_1 = temp32_1 + ADTREG1;
temp32_1 = temp32_1 * 18919;
                                         /* 18919: sqrt(3)/3 << 15 */
temp16_1 = (signed short)(temp32_1 >> 15U);
temp32_1 = (signed long)temp16_1 * (signed long)cos_buf;
temp32_1 = -temp32_1;
temp32_2 = (signed long)temp16_1 * (signed long)sin_buf;
ADTREG0 = (unsigned short)((temp32_0 - temp32_2) >> 14U);
temp32_3 = (signed long)temp16_0 * (signed long)sin_buf;
ADTREG1 = (unsigned short)((temp32_1 - temp32_3) >> 14U);
```



## No.53: Correct typos in the internal calculation formula.

Page: P.1693

Incorrect:

27.3.2.14 Clarke and Park Transformation (Amplitude invariant transformation)

Calculation of PI Control for DC/DC Converter Control:

```
// AAU: PI control for DC/DC converter (Channel.1)
signed long
               temp32 0, temp32 1;
temp32_0 = (signed long)(AL1REF - AIPL1) * (signed long)AKI2;
temp32_1 = (signed long)(ADTRG0 - AL10FS);
temp32_0 = temp32_0 + ((signed long)AL1REF - temp32_1) * (signed long)AKI1;
temp32_0 = temp32_0 + (signed long)ADUTYL1;
// < Overflow/underflow check >
if (temp32_0 > (signed long)(ADUTYMX * 256)) {
   temp32_0 = (signed long)(ADUTYMX * 256);
}
else if (temp32_0 < 0) {
   temp32 \ 0 = 0;
}
AIPL1 = (unsigned short)temp32_1;
ADTREG0 = (unsigned short)temp32_0;
```

Correct:

#### 27.3.2.14 Clarke and Park Transformation (Amplitude invariant transformation)

Calculation of PI Control for DC/DC Converter Control:

```
// AAU: PI control for DC/DC converter (Channel.1)
signed long
               temp32_0, temp32_1;
temp32_0 = (signed long)(AL1REF - AIPL1) * (signed long)AKI2;
temp32_1 = (signed long)(ADTREG0 - AL10FS);
temp32_0 = temp32_0 + ((signed long)AL1REF - temp32_1) * (signed long)AKI1;
temp32_0 = temp32_0 + (signed long)ADUTYL1;
// < Overflow/underflow check >
if (temp32_0 > (signed long)(ADUTYMX * 256)) {
   temp32_0 = (signed long)(ADUTYMX * 256);
}
else if (temp32_0 < 0) {
   temp32_0 = 0;
}
AIPL1 = (unsigned short)temp32_1;
ADTREG0 = (unsigned short)temp32_0;
```



Page: P.170 Incorrect:		he ERADR register.					
Incorrect:	)7						
28.3.3 Int	ernal RAM-ECC Fu	nction					
(1) Error ad	Idress store registe						
	Fi	gure 28-8. Format of Error Address Store Register (ERADR)					
Addres	s: F0200H After re	eset: 0000H <b>R/W</b>					
Symbol	15 14 13	12 11 10 9 8 7 6 5 4 3 2 1 0					
ERADR		ERAD					
	ERAD	Bit error address					
	0000H to FFFFH	Address when a bit error interrupt request is generated					
	2. The r	register value is updated each time a bit error interrupt request is generated.					
Correct: 28.3.3 Int	ernal RAM-ECC Fu	nction					
1) Error ad	Idress store registe	er (ERADR)					
.,		gure 28-8. Format of Error Address Store Register (ERADR)					
	• •	gare 20 0. I official of Error Address of the Register (ERADR)					
		eset: 0000H R					
Symbol	15 14 13	12 11 10 9 8 7 6 5 4 3 2 1 0					
ERADR		ERAD					
	5545	<b></b>					
	ERAD	Bit error address					
	0000H to FFFFH	Address when a bit error interrupt request is generated					
	0000H to FFFFH						
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I						
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					
	0000H to FFFFH Cautions 1. The I	ERADR register should be read by a 16-bit memory manipulation instruction.					



# No.55: Correct typo in register symbols of syndrome code in "Operation Explanation 1".

Page: P.1726

## Incorrect:

## 28.3.5 Code Flash Memory ECC Function

### **Operation Explanation: 1.**

#### **Note** The table below shows the buffer information.

Buffer	Address	Syndrome code
Temporary capture buffer (Code flash bit error detection address register H, Code flash bit error detection address register L)	ERRADRH.ERRADR[19:16], ERRADRL.ERRADR[15:2]	RRADRH ERRDAT[5:0]
Permanent capture buffer (Code flash bit error detection address register n H, Code flash bit error detection address register n L) (n: 1 to 3)	ERRADRnH.ERRADRn[19:16], ERRADRnL.ERRADRn[15:2]	<b>RRADRnH</b> .ERRDATn[5:0]

#### Correct:

## 28.3.5 Code Flash Memory ECC Function

### Operation Explanation: 1.

**Note** The table below shows the buffer information.

Buffer	Address	Syndrome code		
Temporary capture buffer (Code flash bit error detection address register H, Code flash bit error detection address register L)	ERRADRH.ERRADR[19:16], ERRADRL.ERRADR[15:2]	ERRADRH.ERRDAT[5:0]		
Permanent capture buffer (Code flash bit error detection address register n H, Code flash bit error detection address register n L) (n: 1 to 3)	ERRADRnH.ERRADRn[19:16], ERRADRnL.ERRADRn[15:2]	ERRADRnH.ERRDATn[5:0]		



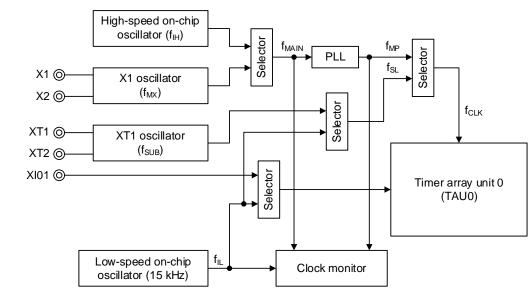
### No.56: Correct typo in pin function name.

#### Page: P.1749

Incorrect: Signal input pin for Timer array unit 0 (TAU0): XI01

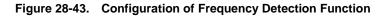
### 28.3.9 Frequency Detection Function

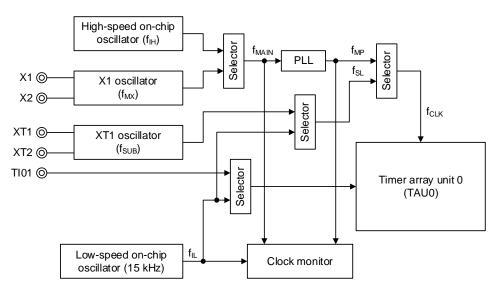




Incorrect: Signal input pin for Timer array unit 0 (TAU0): TI01

## 28.3.9 Frequency Detection Function







### No.57: Add FLPEN bit to the bit setting combination description.

## Page: P.1760

## Incorrect:

## 31.3 Format of On-chip Debug Option Byte

### Figure 31-4. Format of On-chip Debug Option Byte (000C3H/040C3H)

Address: 000C3H/040C3H Note 1 After reset: — (user setting value Note 3)

		•	0	,							
7	6	5	4	3	2	1	0				
OCDENSET	0	FLPEN	0 0 1 HPIEN Note 2 OCDE								
OCDENSET	HPIEN Note 3	OCDERSD	Control of on-chip debug operation								
0	0	0	Disables on-o	hip debug ope	ration.						
1	0	0	Enables on-chip debugging and disables hot plug-in operation.								
			Erases data of flash memory in case of failures in authenticating on-chip debug security ID.								
1	0	1	Enables on-c	hip debugging	and disables h	ot plug-in oper	ation.				
			Does not erase data of flash memory in case of failures in authenticating on-chip debug security ID.								
1	1	1	Enables on-c	hip debugging	and hot plug-ir	n operation.					
			Does not erase data of flash memory in case of failures in authenticating on-chip debug security ID.								
Othe	er than the abo	ve	Setting prohibited								

#### Correct:

## 31.3 Format of On-chip Debug Option Byte

### Figure 31-4. Format of On-chip Debug Option Byte (000C3H/040C3H)

Address: 000C3H/040C3H Note 1 After reset: --- (user setting value Note 3)

7	6	5	4	3	2	1	0
OCDENSET	0	FLPEN	0	0	1	HPIEN Note 2	OCDERSD

FLPEN	OCDENSET	HPIEN Note 2	OCDERSD	Control of on-chip debug operation					
0	Х	Х	Х	Disables flash serial programming and on-chip debugging operation.					
1	0	0	0	Disables on-chip debugging.					
1	1	0	0	<ul> <li>Enables on-chip debugging.</li> </ul>					
				<ul> <li>Disables hot plug-in operation.</li> </ul>					
				<ul> <li>Erases data of flash memory in case of failures in authenticating on-chip debug security ID.</li> </ul>					
1	1	0	1	<ul> <li>Enables on-chip debugging.</li> </ul>					
				<ul> <li>Disables hot plug-in operation.</li> </ul>					
				<ul> <li>Does not erase data of flash memory in case of</li> </ul>					
				failures in authenticating on-chip debug security ID.					
1	1	1	1	<ul> <li>Enables on-chip debugging.</li> </ul>					
				<ul> <li>Enables hot plug-in operation.</li> </ul>					
				<ul> <li>Does not erase data of flash memory in case of</li> </ul>					
				failures in authenticating on-chip debug security ID.					
1	Oth	er than the ab	ove	Setting prohibited					

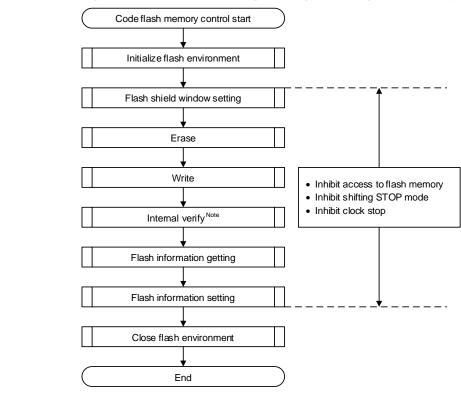


## No.58: Add "HALT mode to prohibited transitions" to flow annotations.

Page: P.1778

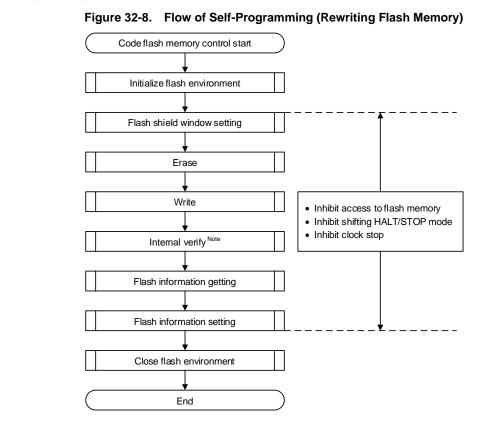
32.7.1 Self-Programming Procedure

#### Figure 32-8. Flow of Self-Programming (Rewriting Flash Memory)



## Correct: Inhibit setting HALT/STOP mode

#### 32.7.1 Self-Programming Procedure





## No.59: Correct description of DCLR bit.

Page: P.1784

Incorrect: Renesas Flash Driver (RFD) provided by Renesas operates correctly.

#### 32.7.2.5 Flash Memory Sequencer Control Register (FSSQ)

#### Figure 32-13. Format of Flash Memory Sequencer Control Register (FSSQ)

DCLR	Operation control of the ECC area sequencer
0	The ECC area sequencer is stopped.
1	The ECC area sequencer is started.

Correct:

### 32.7.2.5 Flash Memory Sequencer Control Register (FSSQ)

#### Figure 32-13. Format of Flash Memory Sequencer Control Register (FSSQ)

DCLR	Operation stop control of the ECC area sequencer
0	The ECC area sequencer is operating.
1	The ECC area sequencer is stopped.

## No.60: Correct the typo in the description of the DCLR bit of the FSSQ register.

Page: P.1791

### Incorrect: Renesas Flash Driver (RFD) provided by Renesas operates correctly.

### 32.7.2.12 Flash Memory Sequencer Control Register (FSSQ)

The FLWE register stores the ECC data which is used at flash memory programming when the DCLR bit is 1 in the FSSQ register. The lower 6-bit data is used at code flash memory programming, and the lower 4-bit data is used at data flash memory programming.

#### Correct:

## 32.7.2.12 Flash Memory Sequencer Control Register (FSSQ)

When the DCLR bit of the FSSQ register is 0, the FLWE register stores ECC data used in flash memory programming. When the DCLR bit is 1, any ECC programing data can be written for ECC diagnosis.

The lower 6-bit data is used at code flash memory programming, and the lower 4-bit data is used at data flash memory programming.

## No.61: Add "HALT mode to prohibited transitions" to bullet point description.

Page: P.1816

Incorrect:

## 32.9.1 Overview of the Data Flash Memory

• Transition to the STOP status is prohibited while rewriting the data flash memory.

#### Correct:

## 32.9.1 Overview of the Data Flash Memory

• Transition to the HALT/STOP status is prohibited while rewriting the data flash memory.



## No.62: Correct a typo in the operation description of the OR1 instruction.

# Page: P.1843

Incorrect:

Instruction	Mnemonic	Operands	Bytes	Clo	cks	Operation	Flag		
Group				Note 1	Note 2		Z	AC	CY
Bit	OR1	CY, A.bit	2	1	-	$CY \leftarrow CY \lor A.bit$			×
manipulate		CY, PSW.bit	3	1	-	CYX.←.CY.y.PSW.bit			×
		CY, saddr.bit	3	1	-	$CY \leftarrow CY \lor (saddr).bit$			×
		CY, sfr.bit	3	1	-	$CY \leftarrow CY \lor sfr.bit$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \lor (HL).bit$			×
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \lor (ES, HL).bit$			×

## Table 35-6. Operation List (14/18)

#### Correct:

Table 35-6.	Operation	List (14/18)
-------------	-----------	--------------

Instruction	Mnemonic	Operands	Bytes	Clo	cks	Operation		Flag	
Group				Note 1	Note 2		Z	AC	CY
Bit	OR1	CY, A.bit	2	1	-	$CY \leftarrow CY \lor A.bit$			×
manipulate		CY, PSW.bit	3	1	-	$CY \leftarrow CY \lor PSW.bit$			×
		CY, saddr.bit	3	1	-	$CY \leftarrow CY \lor (saddr).bit$			×
		CY, sfr.bit	3	1	-	$CY \leftarrow CY \lor sfr.bit$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \lor (HL).bit$			×
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \lor (ES, HL).bit$			×

## No.63, 65, 67: Correct "Note 4" description below the table.

Page: P.1899, P.1953, P.2005 Incorrect:

## 36.10, 37.10, 38.10 Flash Memory Programming Characteristics

Note 4. The specified data retention time is given under the condition that the average temperature (TA) is 85°C. or below.

**Correct:** Simplify the explanation of retention temperature.

36.10, 37.10, 38.10 Flash Memory Programming Characteristics

**Note 4.** The average temperature for data retention.



## No.64, 66, 68: Add "(1) Code flash memory processing time" and "(2) Data flash memory processing

## time" sections.

Page: P.1899, P.1953, P.2005

Correct:

## (1) Code flash memory processing time

Item		fclк = 2 MHz		fclк = 4 MHz		fclк = 8 MHz		fclк <b>= 16 MHz</b>		Unit
	Conditions	TYP.	MAX.	TYP.	MAX.	TYP.	MAX.	TYP.	MAX.	
Programming time	4 bytes	74.0	690.0	61.0	580.0	56.0	530.0	54.0	510.0	μs
Erasure time	1 KB	6.9	245.0	6.1	230.0	5.8	225.0	5.6	220.0	ms
Blank checking time	4 bytes	-	29.0	_	22.0	_	19.0	-	17.0	μs
	1 KB	-	800.0	-	405.0	-	245.0	-	145.0	μs
Internal verify time	4 bytes	_	350.0	_	175.0	_	90.0	_	45.0	μs
	1 KB	Ι	19.0	_	9.5	-	5.0	Ι	2.5	ms

Item	_	fськ = 2	20 MHz	fськ = 3	32 MHz	fclk = 4	Unit	
	Conditions	TYP.	MAX.	TYP.	MAX.	TYP.	MAX.	
Programming time	4 bytes	54.0	510.0	53.0	500.0	53.0	500.0	μs
Erasure time	1 KB	5.6	220.0	5.5	220.0	5.5	220.0	ms
Blank checking time	4 bytes	-	17.0	-	16.0	-	16.0	μs
	1 KB	-	145.0	-	135.0	-	135.0	μs
Internal verify time	4 bytes	_	35.0	_	22.0	_	18.0	μs
	1 KB	_	2.0	_	1.2	-	1.0	ms

Caution The listed values do not include the time until the operations of the flash memory start following execution of an instruction by software.

## (2) Data flash memory processing time

Item		fclк = 2 MHz		fclк = 4 MHz		fclк = 8 MHz		fclк = 16 MHz		Unit
	Conditions	TYP.	MAX.	TYP.	MAX.	TYP.	MAX.	TYP.	MAX.	
Programming time	1 byte	60.0	550.0	49.0	450.0	44.0	410.0	42.0	390.0	μs
Erasure time	1 KB	11.5	340.0	8.4	275.0	7.1	250.0	6.3	235.0	ms
Blank checking time	1 byte	-	29.0	_	22.0	_	19.0	-	17.0	μs
	1 KB	Ι	3.1	_	1.6	_	0.95	-	0.55	ms
Internal verify time	1 byte	_	350.0	_	175.0	-	90.0	_	45.0	μs
	1 KB	_	76.0	_	38.0	-	19.0	_	9.5	ms

Item		fclк = 20 MHz		fclк = 32 MHz		fclк = 40 MHz		Unit
	Conditions	TYP.	MAX.	TYP.	MAX.	TYP.	MAX.	
Programming time	1 byte	42.0	390.0	41.0	380.0	41.0	380.0	μs
Erasure time	1 KB	6.3	235.0	6.2	235.0	6.2	235.0	ms
Blank checking time	1 byte	_	17.0	-	16.0	-	16.0	μs
	1 KB	-	0.55	-	0.5	-	0.5	ms
Internal verify time	1 byte	_	35.0	_	22.0	_	18.0	μs
	1 KB	_	7.5	-	4.7	-	3.8	ms

Caution The listed values do not include the time until the operations of the flash memory start following execution of an instruction by software.

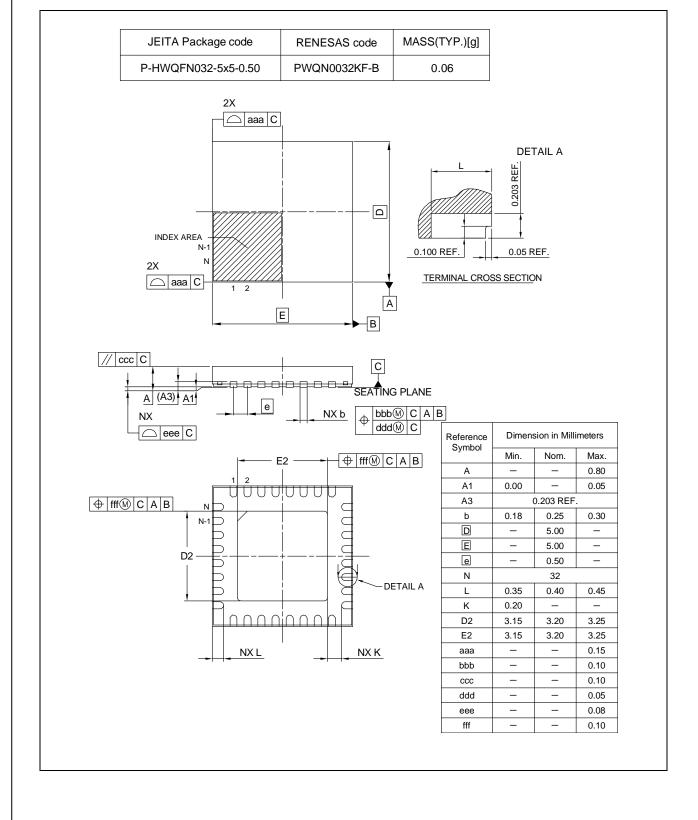


## No.69: Correct the value of the pin cross section in the figure.

Page: P.2007

## Incorrect: Terminal cross section: [0.100 REF, 0.05 REF, 0.203 REF], A3: 0.203 REF

## 39.1 32-pin products





#### Correct: Terminal cross section: [0.06 REF, 0.05 REF, 0.20 REF], A3: 0.20 REF

## 39.1 32-pin products

