Date: Dec. 3, 2024

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 User's Manual: Hardware Rev.1.00		Information Category	Technical Notification		
Applicable Product	RL78/F23, F24 Group	Lot No. All lots	Reference Document	RL78/F23, F24 User's Manual: Hardware Rev.1.00 (R01UH0944EJ0100)		

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

(2/3)

No	Corrections	R01UH0944EJ0100	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) Data flash memory processing time” sections.	P.1899	P.57
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) Data flash memory processing time” sections.	P.1953	P.57
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) Data flash memory processing time” sections.	P.2005	P.57
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)

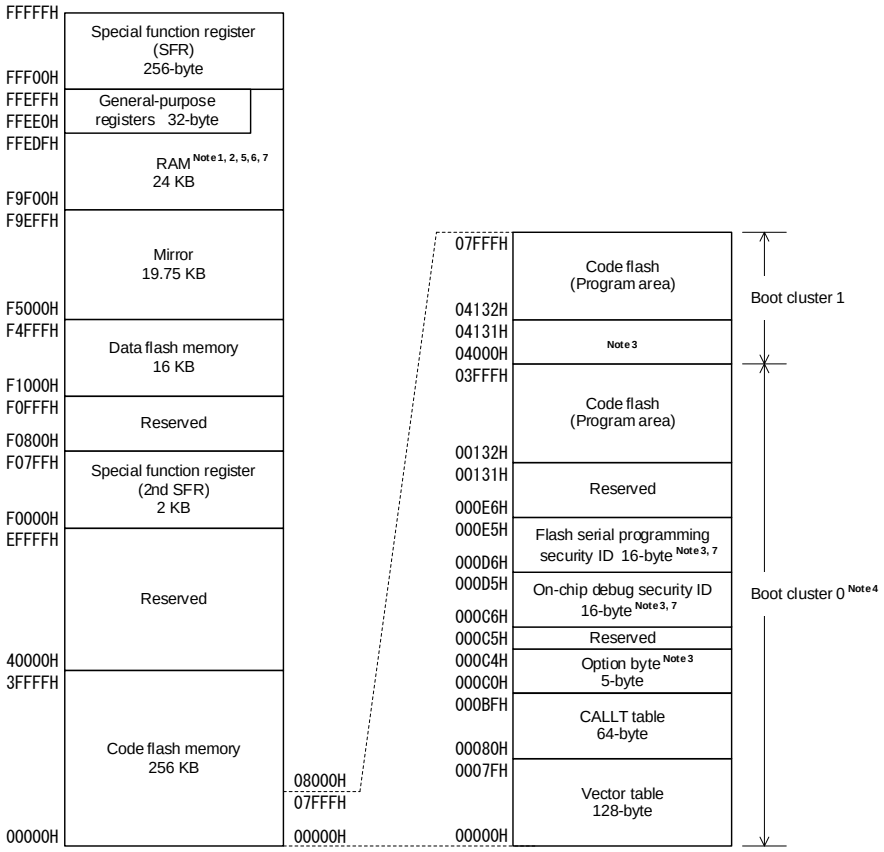
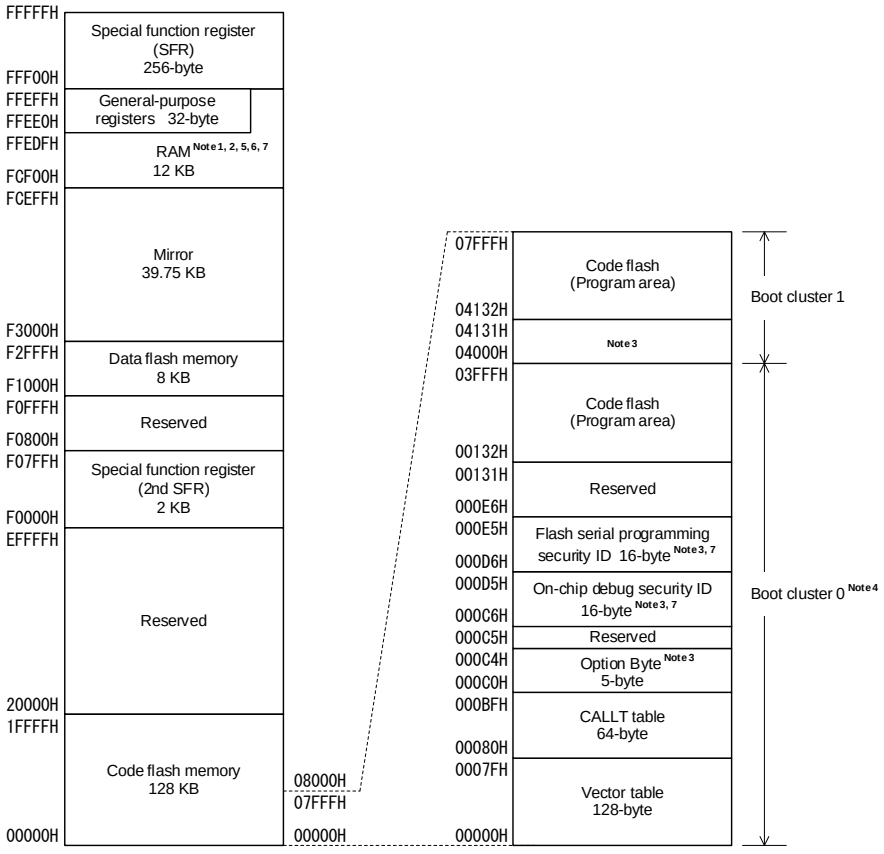


Figure 3-2. Memory Map (RL78/F23)



Correct: Note number of Security ID area (000C6H to 000E5H): Note 3, 8

Figure 3-1. Memory Map (RL78/F24)

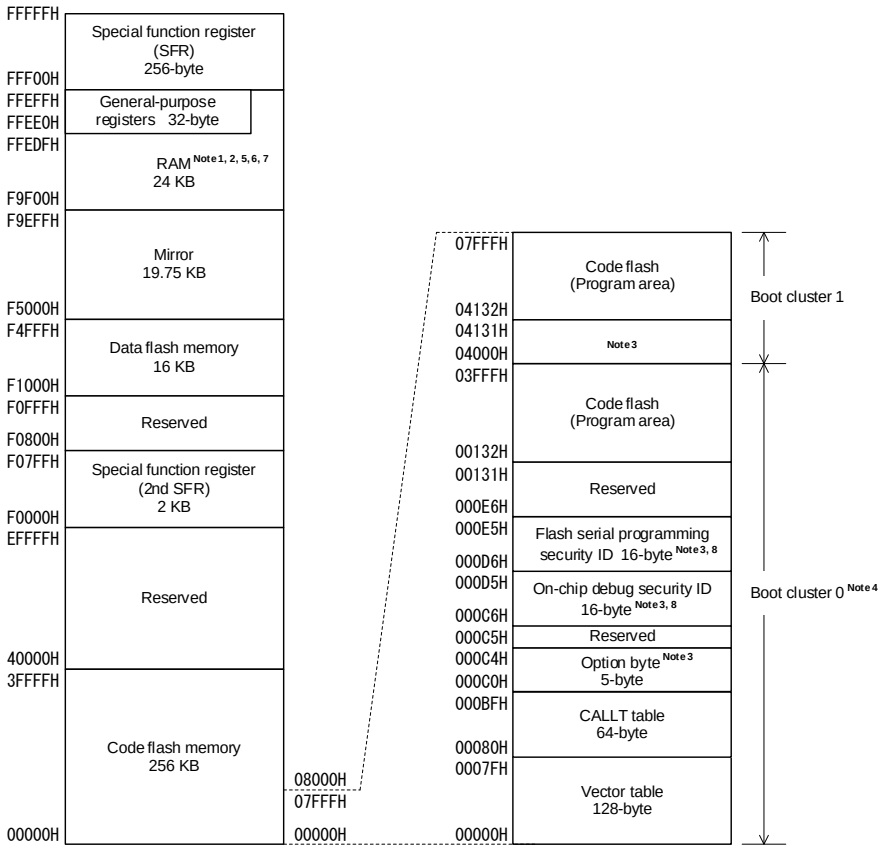
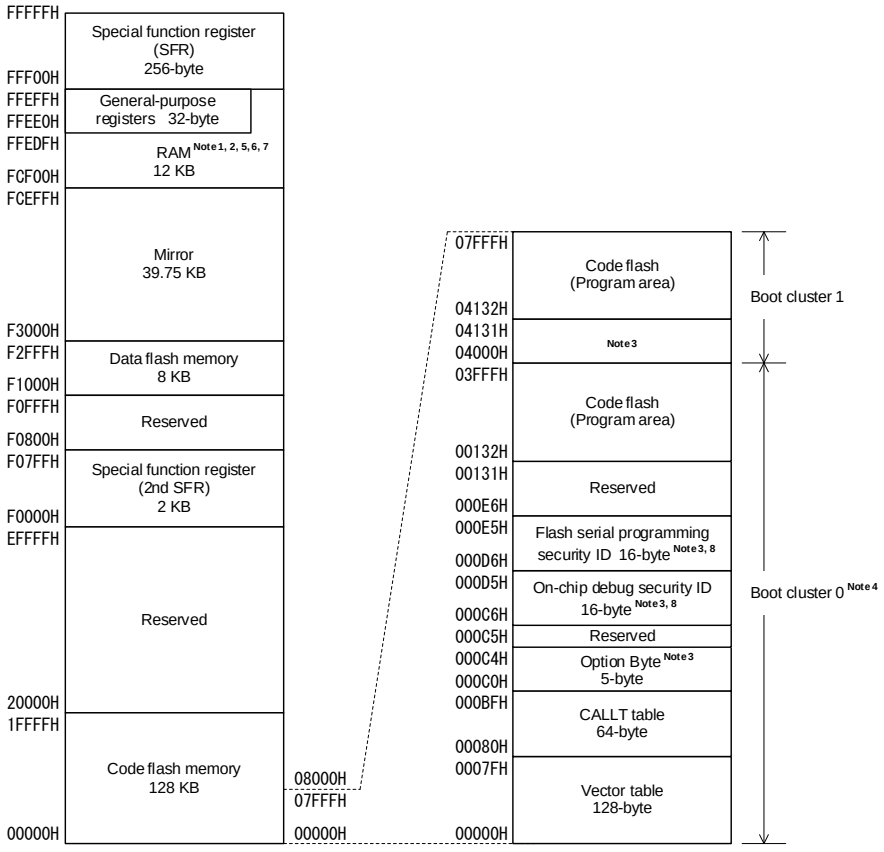


Figure 3-2. Memory Map (RL78/F23)



No.2: Correct the typo in the reserved word definition for PMS0 bit.

Page: P.263

Incorrect:

Figure 4-96. Port Mode Select Register (PMS)

Address: F0077H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	<0>
PMS	0	0	0	0	0	0	0	PMS0

Correct:

Figure 4-96. Port Mode Select Register (PMS)

Address: F0077H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PMS	0	0	0	0	0	0	0	PMS0

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

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

Correct:

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

Pin Name	Alternate Function		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, PIOR93=1	0/1	0	0	1	—	×
P125	(LRXD1)	Input	PIOR45=1, PIOR93=1	—	0	1	×	0/1	0/1

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 (PIOR42 = 1)	
SSI10 Note	P54 (PIOR42 = 0)	P77 (PIOR42 = 1)	
SCK11 Note	P71 (PIOR43 = 0)		P153 (PIOR43 = 1)
SI11 Note	P70 (PIOR43 = 0)		P152 (PIOR43 = 1)
SO11	P72 (PIOR43 = 0, PIOR92 = 0)	P32 (PIOR43 = 0, PIOR92 = 1)	P151 (PIOR43 = 1)
SSI11 Note	P73 (PIOR43 = 0)		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

Incorrect: Starting point of the arrow: TAU stop → 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 TImn 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.

Operation is resumed.

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 → 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 TImn 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.

Operation is resumed.

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 controls the operation of the counter.

Be sure to write “1” to it to read or write the counter value.

As the internal counter (16 bits) is continuing to run, complete reading or writing within one second and turn back to 0.

When RWAIT = 1, it takes up to 1 operating clock (f_{RTC}) until the counter value can be read or written (RWST = 1).

Notes 1, 2

When reading 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, complete the processing from setting the RWAIT bit to 1 to setting it to 0 before generation of the next constant-period interrupt.

When the internal counter (16 bits) overflowed while RWAIT = 1, it keeps the event of overflow until RWAIT = 0, then counts up.

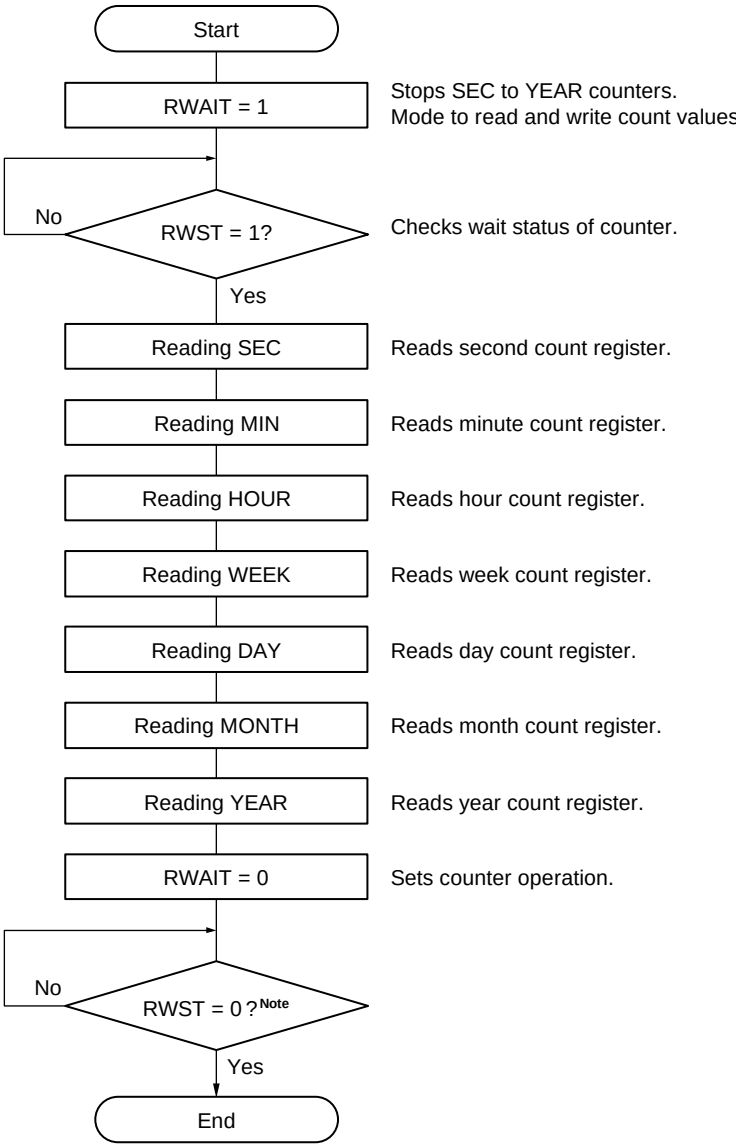
However, when it wrote a value to second count register, it will not keep the overflow event.

Note 7: Change the description of the “Caution” below the figure.

Page: P.684

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

Figure 9-23. Procedure for Reading Real-time Clock



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.

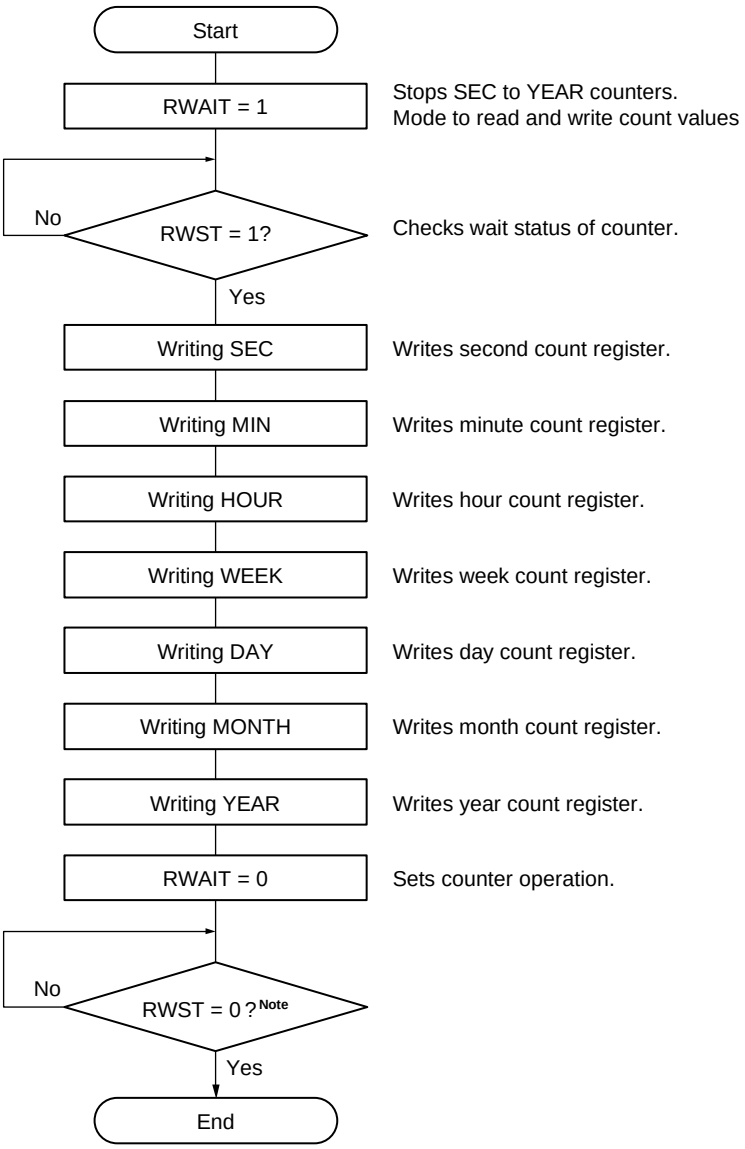
Also, it is not necessary to read all the registers, and only some of the registers may be read.

No.8: Change the description of the “Caution 1” below the figure.

Page: P.685

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

Figure 9-24. Procedure for Writing Real-time Clock



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.

No.9: Add the Note 1 to the ADNDIS[4] bit and Note 2 to the ADNDIS[3:0] bits, and listed the notes separately.

Page: P.729

Incorrect:

Figure 12-16. Format of A/D Disconnection Detection Control Register (ADDISCR)

Address: F06BAH (ADWINR=07H) After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ADDISCR	0	0	0	ADNDIS[4:0]				

ADNDIS4	Setting of precharge/discharge selection
0	Discharge select
1	Precharge select
The ADNDIS4 bit selects either precharge or discharge for the A/D disconnection detection assist function. The ADNDIS4 bit should be set while the ADCSR.ADST bit is 0.	

ADNDIS[3:0]	Period of precharge/discharge
0000B	Disables the disconnection detection assist function.
0001B	Setting prohibited.
0010B	2 cycles (2/ADCLK)
0011B	3 cycles (3/ADCLK)
0100B	4 cycles (4/ADCLK)
0101B	5 cycles (5/ADCLK)
0110B	6 cycles (6/ADCLK)
0111B	7 cycles (7/ADCLK)
1000B	8 cycles (8/ADCLK)
1001B	9 cycles (9/ADCLK)
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)
When the self-diagnosis is used, the disconnection detection assistance cannot be used. In that case, the ADNDIS[3:0] bits should be set to 0000B. When the ADNDIS[3:0] bits are set to any values other than 0000B or 0001B, and the disconnection detection assistance is enabled, the disconnection detection assistance for the channel-dedicated sample-and-hold circuit used for analog inputs are also enabled. ADNDIS[3:0] bits should be set while the ADCSR.ADST bit is 0.	

Note When the internal reference voltage is converted, AD 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.

Correct:

Figure 12-16. Format of A/D Disconnection Detection Control Register (ADDISCR)

Address: F06BAH (ADWINR=07H) After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ADDISCR	0	0	0	ADNDIS[4:0]				

ADNDIS4	Setting of precharge/discharge selection Note 1
0	Discharge select
1	Precharge select
The ADNDIS4 bit selects either precharge or discharge for the A/D disconnection detection assist function. The ADNDIS4 bit should be set while the ADCSR.ADST bit is 0.	

ADNDIS[3:0]	Period of precharge/discharge Note 2
0000B	Disables the disconnection detection assist function.
0001B	Setting prohibited.
0010B	2 cycles (2/ADCLK)
0011B	3 cycles (3/ADCLK)
0100B	4 cycles (4/ADCLK)
0101B	5 cycles (5/ADCLK)
0110B	6 cycles (6/ADCLK)
0111B	7 cycles (7/ADCLK)
1000B	8 cycles (8/ADCLK)
1001B	9 cycles (9/ADCLK)
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)
When the self-diagnosis is used, the disconnection detection assistance cannot be used. In that case, the ADNDIS[3:0] bits should be set to 0000B. When the ADNDIS[3:0] bits are set to any values other than 0000B or 0001B, and the disconnection detection assistance is enabled, the disconnection detection assistance for the channel-dedicated sample-and-hold circuit are also enabled. ADNDIS[3:0] bits should be set while the ADCSR.ADST bit is 0.	

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

2. 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 above	Setting prohibited.
The ADPAGE bits 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 above	Setting prohibited.
The ADPAGE bits 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.	

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)

Address: F02A0H After reset: 00H

Symbol	<7>	6	5	4	3	2	<1>	0
CMPCTL	HCMPON	CDFS1	CDFS0	CEGN	CEGP	CSTEN	COE	CINV

CSTEN	STOP mode release enable Notes 5, 6	R/W
0	Releasing STOP mode by comparator interrupt disabled	R/W
1	Releasing STOP mode by comparator interrupt enabled	

Note 5. To enable releasing STOP mode by the comparator interrupt, **set this bit to 0** and also set bits CDFS1, CDFS0, and CINV to 00B (noise filter not used).

Correct:

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

Address: F02A0H After reset: 00H

Symbol	<7>	6	5	4	3	2	<1>	0
CMPCTL	HCMPON	CDFS1	CDFS0	CEGN	CEGP	CSTEN	COE	CINV

CSTEN	STOP mode release enable Notes 5, 6	R/W
0	Releasing STOP mode by comparator interrupt disabled	R/W
1	Releasing STOP mode by comparator interrupt enabled	

Note 5. To enable releasing STOP mode by the comparator interrupt, set this bit to 1 and also set bits CDFS1, CDFS0, and CINV to 00B (noise filter not used).

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

No.13: Correct the typo in the register symbol in “Caution 2”.

Page: P.815

Incorrect:

Figure 15-9. Format of Serial Flag Clear Trigger Register mn (SIRmn)

Address: F0104H, F0105H (SIR00), F0106H, F0107H (SIR01), F0144H, F0145H (SIR10), F0146H, F0147H (SIR11) After reset: 0000H R/W

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	FECTmn	PECTmn	OVCTmn

Cautions 1. Be sure to clear bits 15 to 3 to 0.

2. Use the SIRmn register to clear only the error flag set in the SSRn 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.

Correct:

Figure 15-9. Format of Serial Flag Clear Trigger Register mn (SIRmn)

Address: F0104H, F0105H (SIR00), F0106H, F0107H (SIR01), F0144H, F0145H (SIR10), F0146H, F0147H (SIR11) After reset: 0000H R/W

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	FECTmn	PECTmn	OVCTmn

Cautions 1. Be sure to clear bits 15 to 3 to 0.

2. 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.

No.14: Correct the typo in the bit symbol in “Caution”.

Page: P.816

Incorrect:

Figure 15-10. Format of Serial Status Register mn (SSRmn) (1/2)

Address: F0100H, F0101H (SSR00), F0102H, F0103H (SSR01), F0140H, F0141H (SSR10), F0142H, F0143H (SSR11) After reset: 0000H R

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 is written to the SDRmn register when BFFmn = 1, the transmit/receive data stored in the register is discarded and an overrun error (OVEmn = 1) is detected.

Correct:

Figure 15-10. Format of Serial Status Register mn (SSRmn) (1/2)

Address: F0100H, F0101H (SSR00), F0102H, F0103H (SSR01), F0140H, F0141H (SSR10), F0142H, F0143H (SSR11) After reset: 0000H R

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 is written to the SDRmn register when BFFmn = 1, the transmit/receive data stored in the register is discarded and an overrun error (OVFmn = 1) is detected.

No.15: Add the description of serial data input/output (SDAmn) to the text. Also add the setting example when using the SDAmn function.

Page: P.829

Correct:

15.3.19 Port Mode Registers (PM1, PM3 to PM7, PM12)

These registers set input/output of ports1, 3 to 7, and 12 in 1-bit units.

When using the serial data output or serial clock output, or when using serial data input/output (SDAmn), set the bit in the port mode register (PMmn) corresponding to each port to 0. And set the bit in the port register (Pmn) corresponding to each port to 1.

Also, when using the serial data input or serial clock input, set the PMmn bit corresponding to each port to 1.

At this time, the bit in the Pmn bit may be 0 or 1.

Example 1: When using P12/**SO10**/**TXD1** for serial data output

Set the PM12 bit to 0.

Set the P12 bit of the P1 register to 1.

Example 2: When using P11/SI10/**SDA10**/RXD1 for serial data input/output

Set the PM11 bit to 0.

Set the P11 bit of the P1 register to 1.

Example 3: When using P16/**SI00**/SDA00/**RXD0** for serial data input

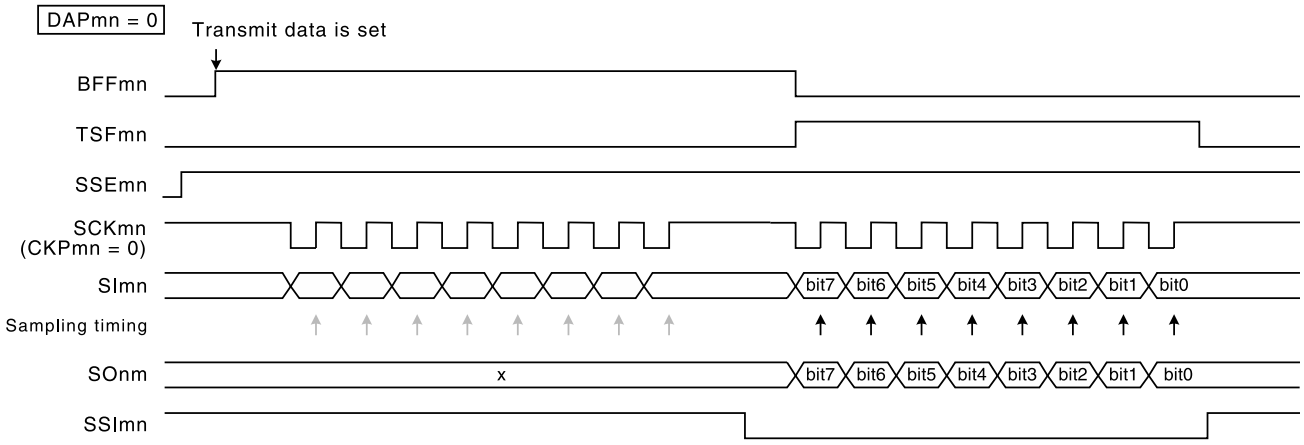
Set the PM16 bit to 1.

No.16: Correct a typo in SOmn output.

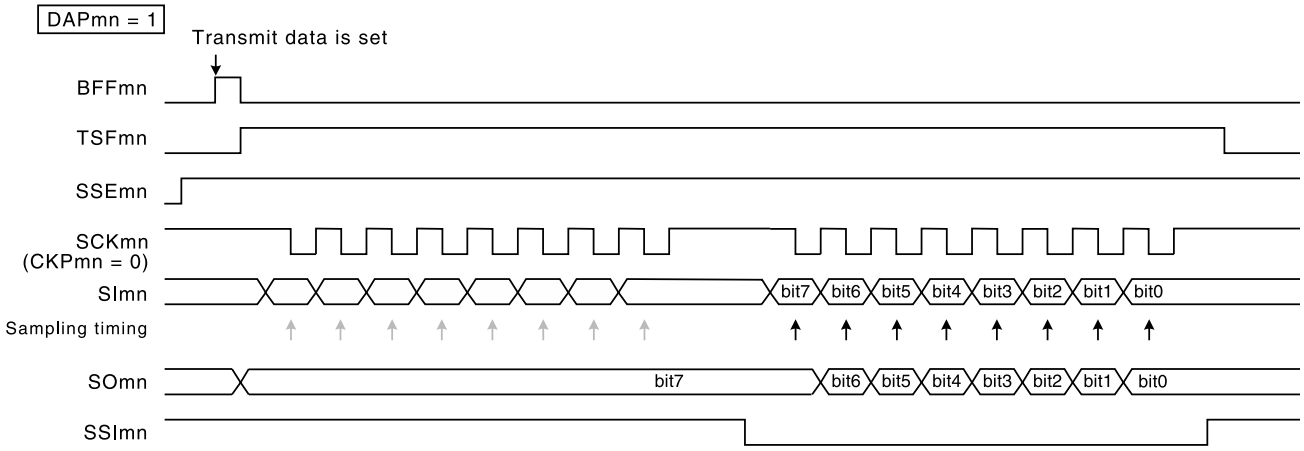
Page: P.901

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



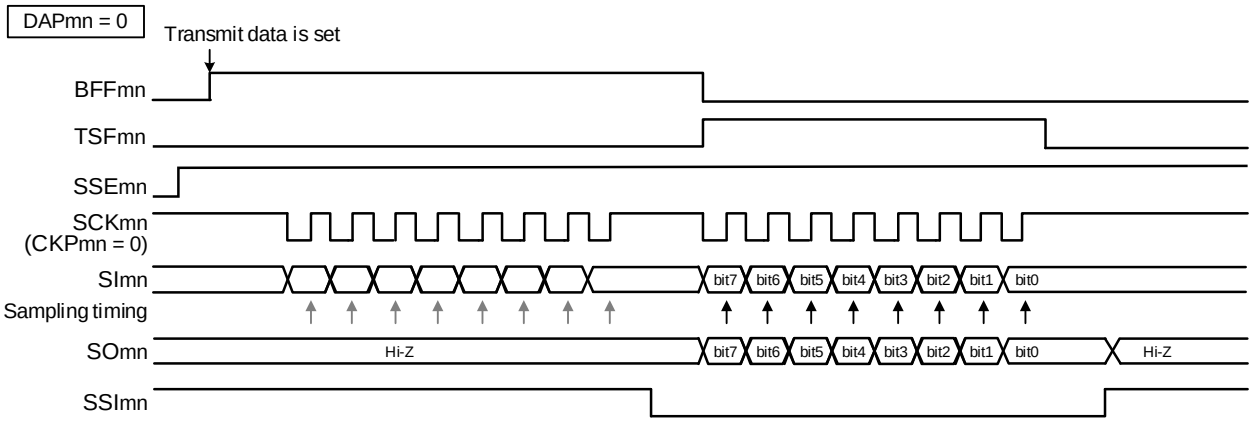
While SSImn is at high level, transmission is not performed even if the falling edge of SCKmn (serial clock) arrives, and neither is receive data sampled in synchronization with the rising edge.
When SSImn goes to low level, data is output (shifted) in synchronization with the falling edge of the serial clock and a reception operation is performed in synchronization with the rising edge.



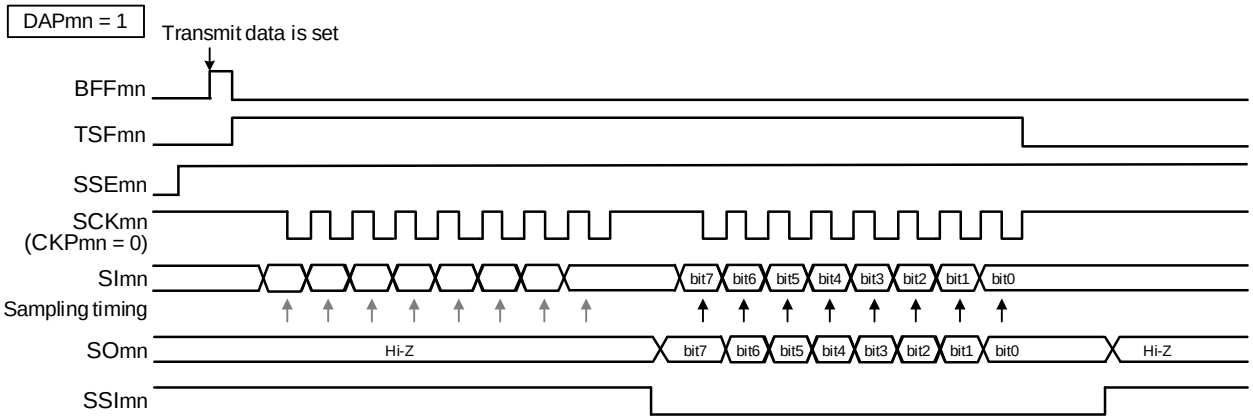
If DAPmn = 1, when transmit data is set while SSImn is at high level, the first data (bit 7) is output to the data output. However, no shift operation is performed even if the rising edge of SCKmn (serial clock) arrives, and neither is receive data sampled in synchronization with the falling edge. When 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.

Correct: SOmn pin output changes according to the input on the SSImn pin.

Figure 15-75. Slave Select Input Function Timing Diagram



While SSImn is at high level, transmission is not performed even if the SCKmn (serial clock) arrives, and neither is the 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.



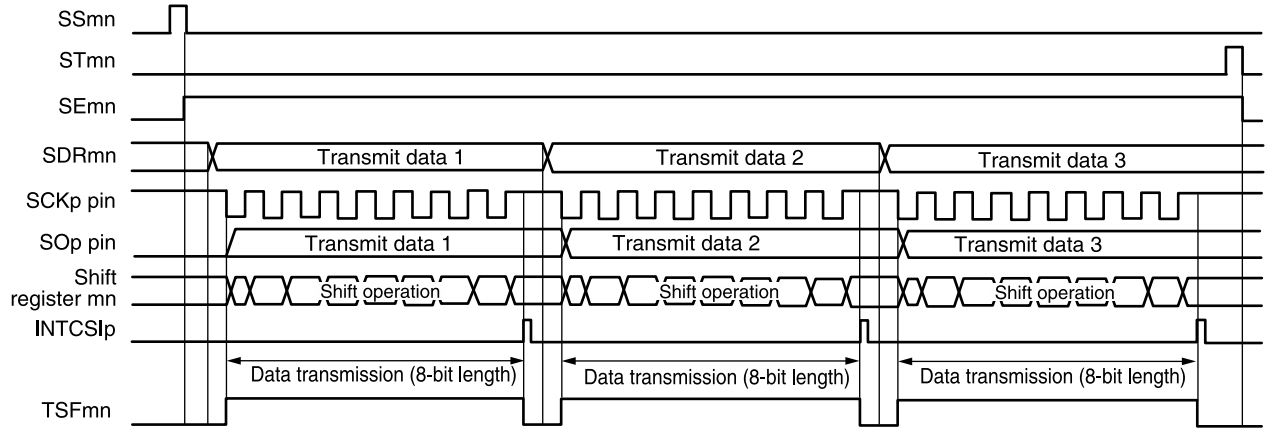
While SSImn is at high level, transmission is not performed even if the SCKmn (serial clock) arrives, and neither is the receive data sampled in synchronization with the falling edge.
When SSImn goes to low level, data is output (shifted) in synchronization with the rising edge of the serial clock and a reception operation is performed in synchronization with the falling edge.

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

Page: P.908

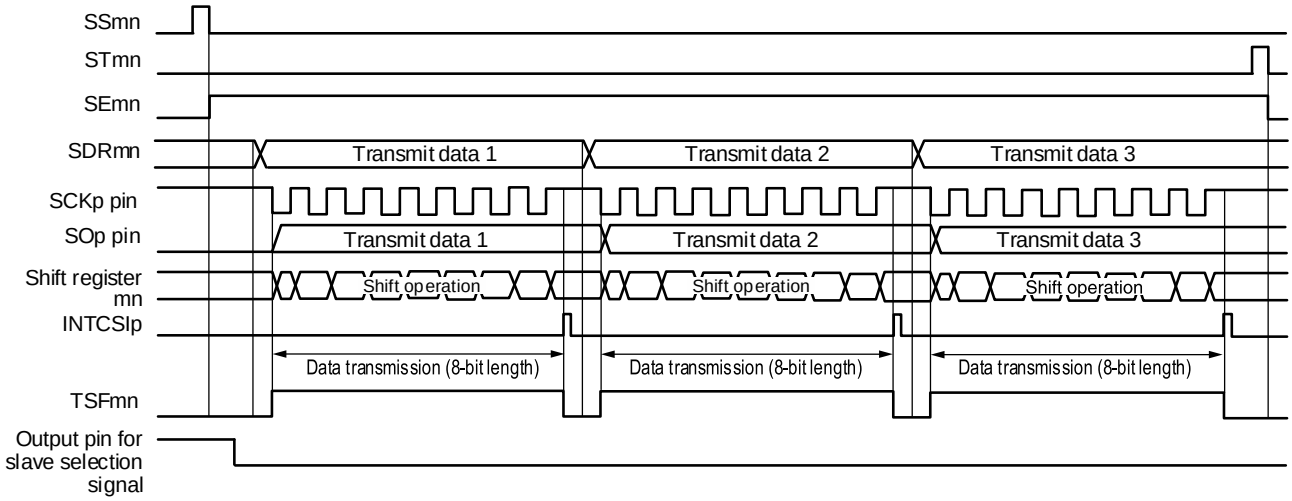
Incorrect:

Figure 15-80. Timing Chart of Master Transmission (in Single-Transmission Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



Correct: For master transmission, add “Output pin for slave selection signal” to the figure.

Figure 15-80. Timing Chart of Master Transmission (in Single-Transmission Mode)
(Type 1: DAPmn = 0, CKPmn = 0)

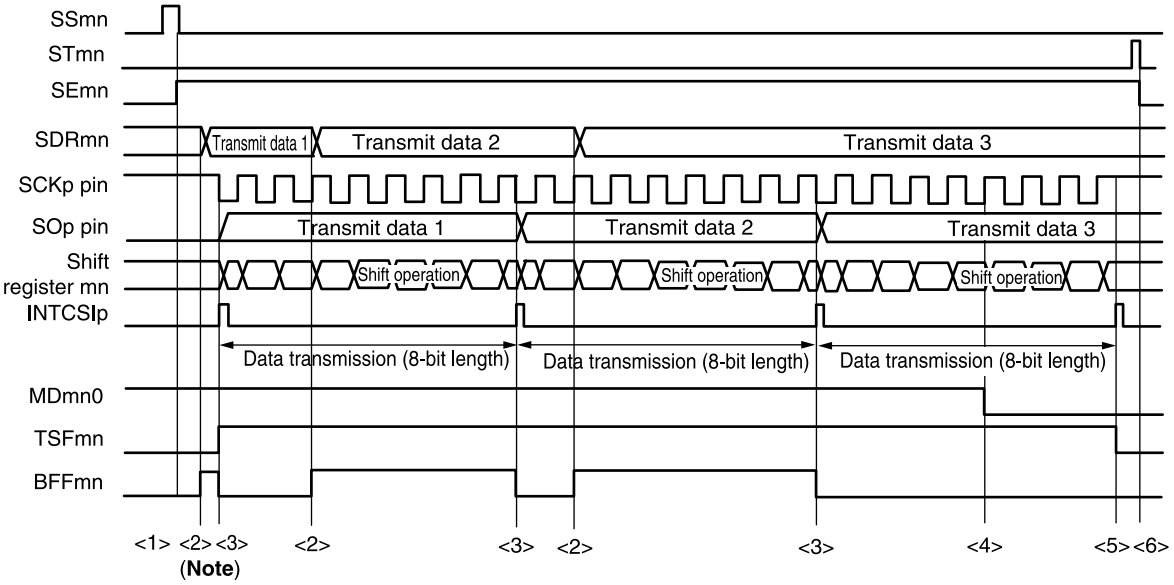


No.18: Add slave selection signal output pin to the figure.

Page: P.910

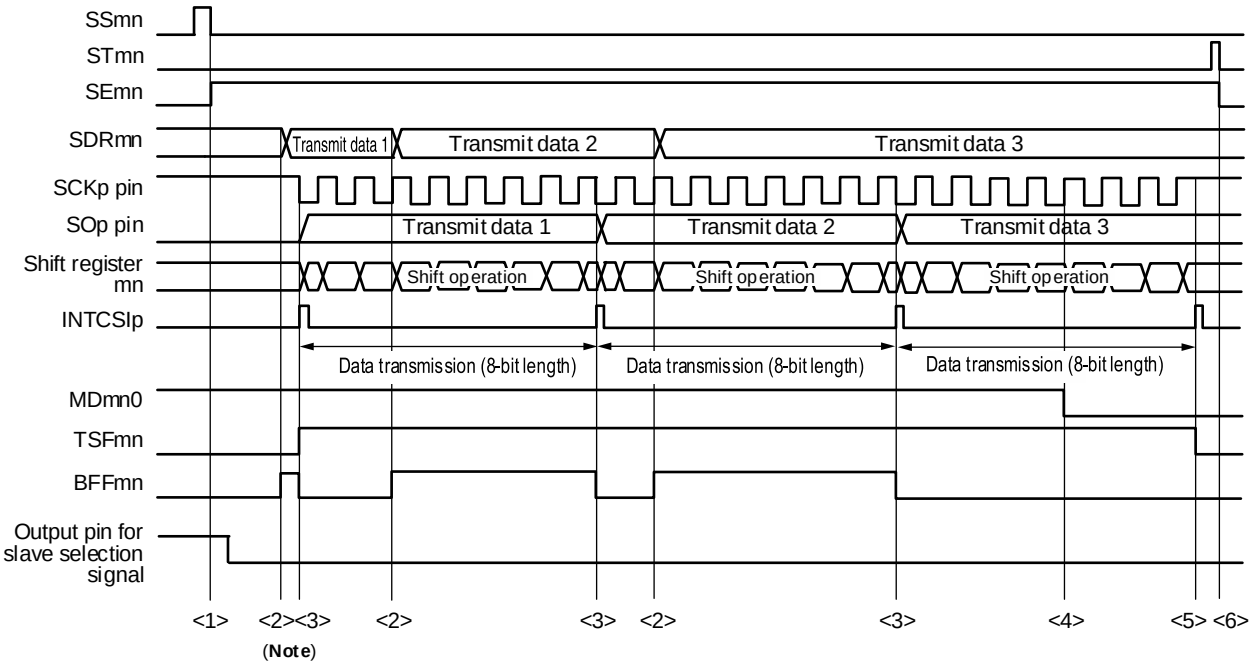
Incorrect:

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



Correct: For master transmission, add “Output pin for slave selection signal” to the figure.

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

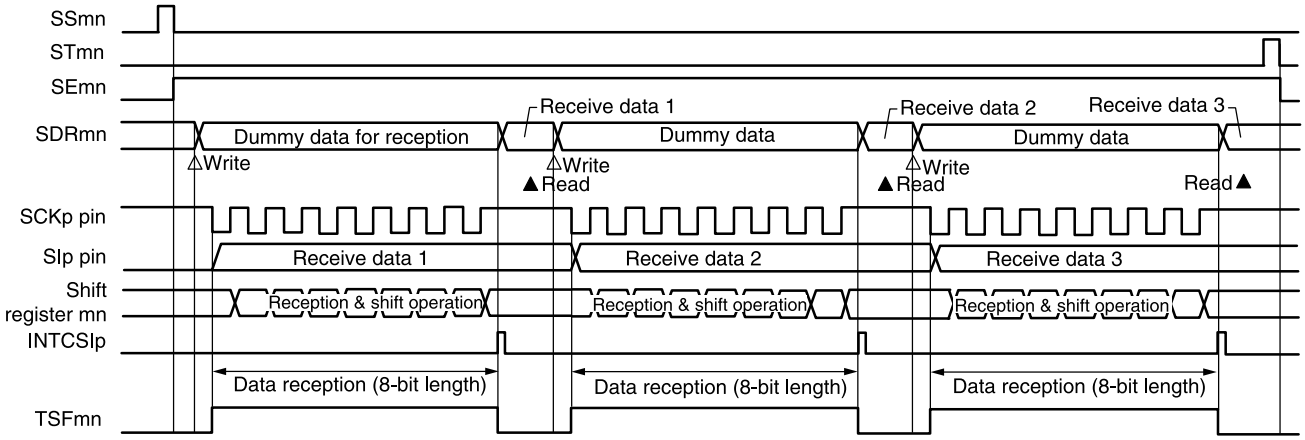


No.19: Add the slave selection signal output pin to the figure.

Page: P.917

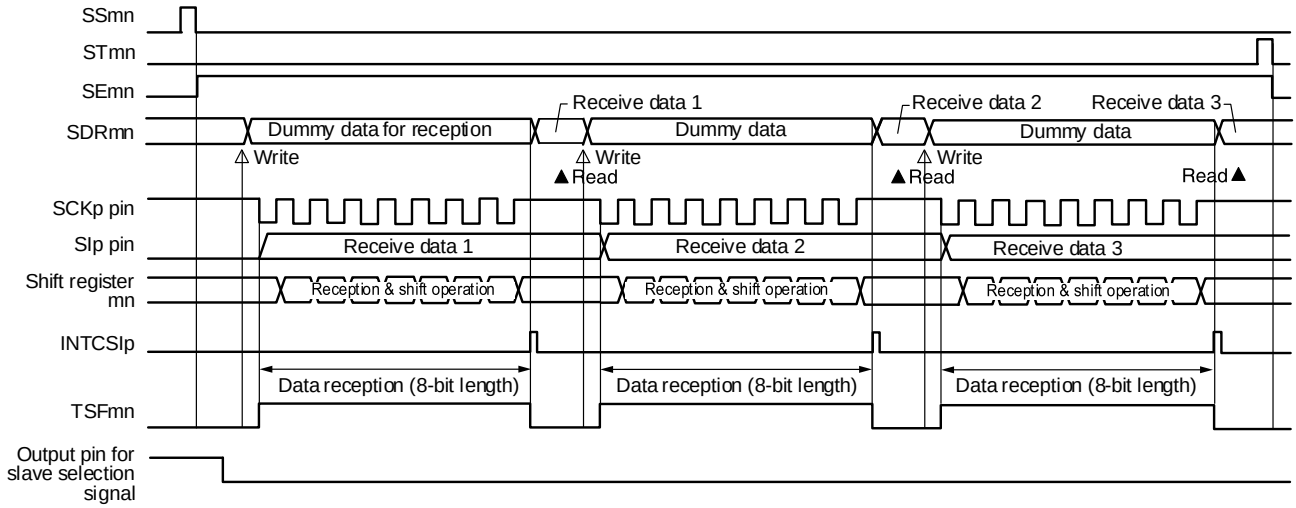
Incorrect:

Figure 15-88. Timing Chart of Master Reception (in Single-Reception Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



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)

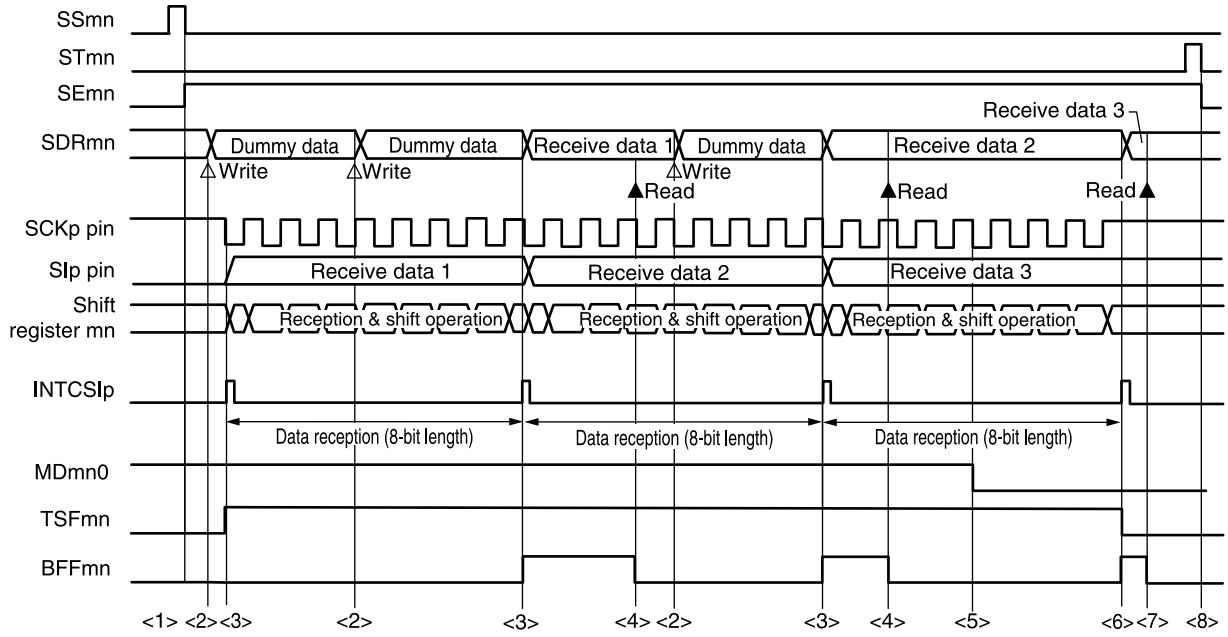


No.20: Add the slave selection signal output pin to the figure.

Page: P.919

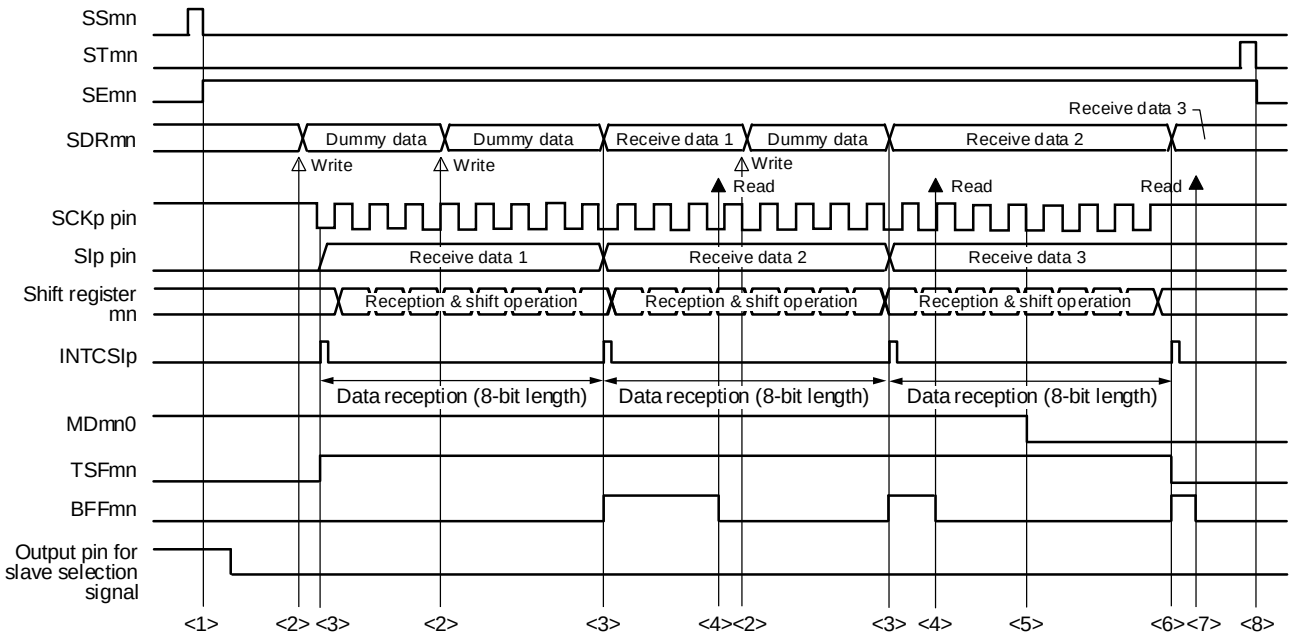
Incorrect:

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



Correct: For master reception, add “Output pin for slave selection signal” to the figure.

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

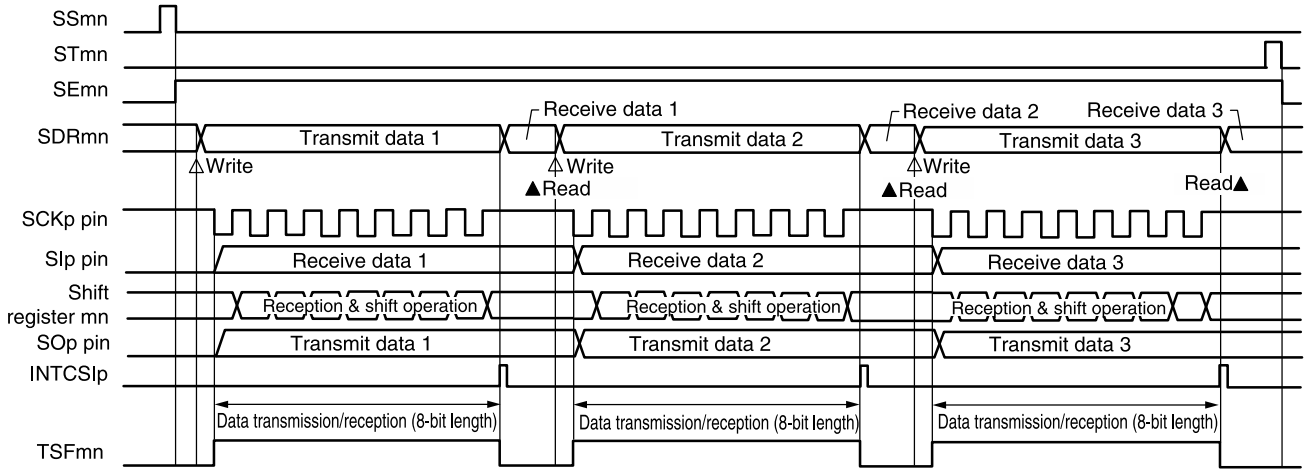


No.21: Add the slave selection signal output pin to the figure.

Page: P.928

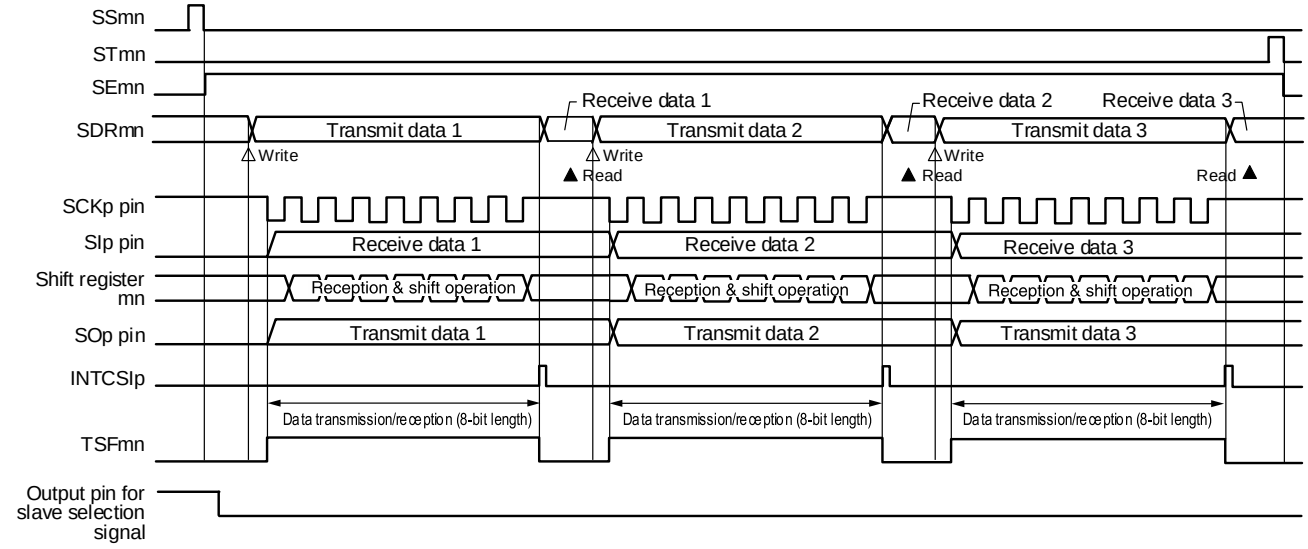
Incorrect:

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.

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

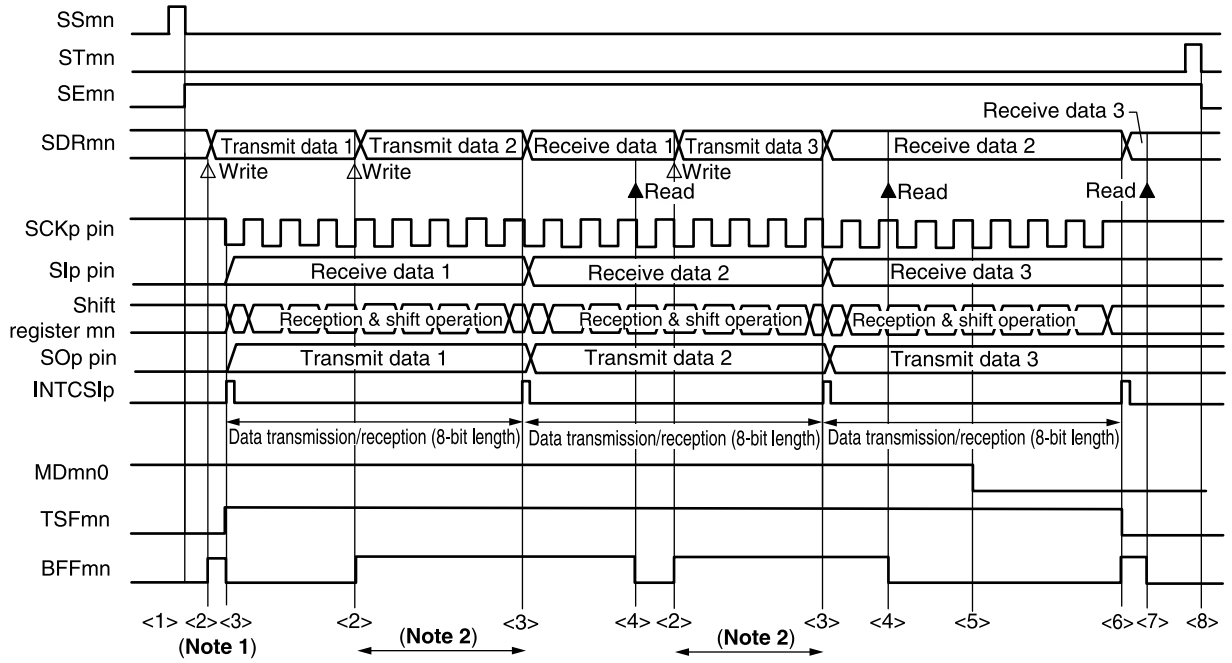


No.22: Add the slave selection signal output pin to the figure.

Page: P.930

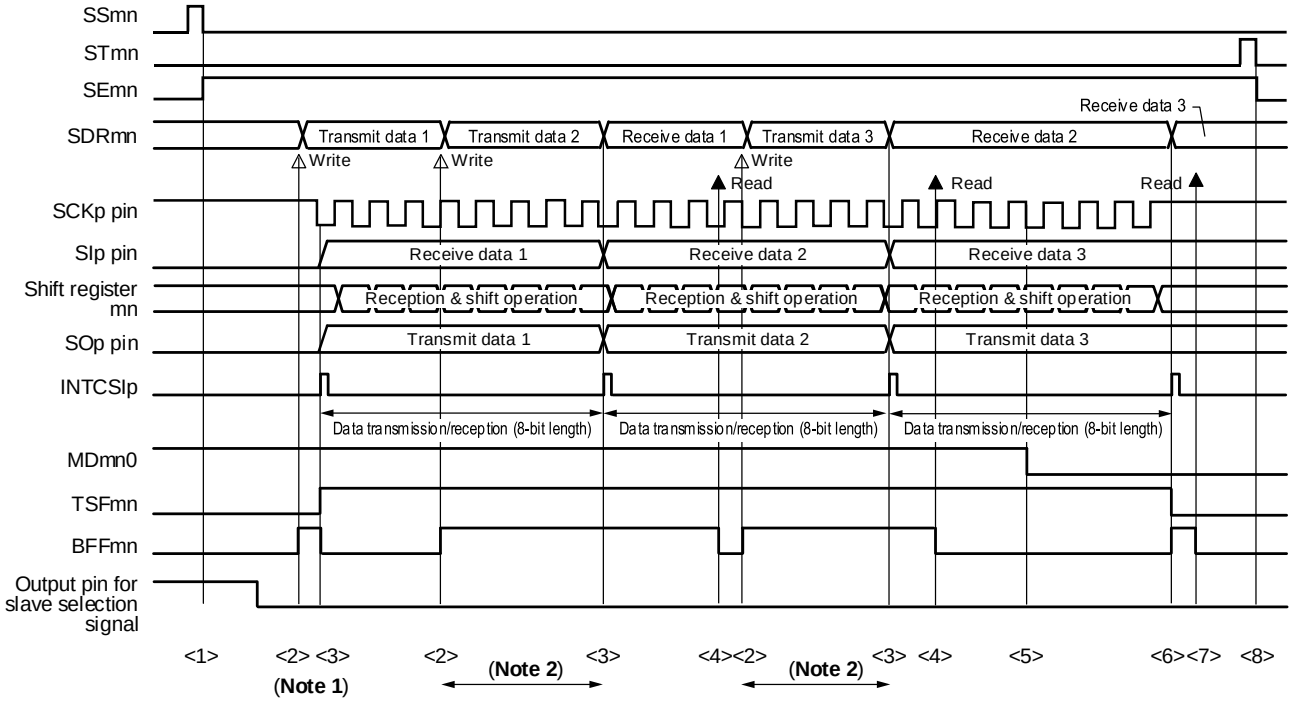
Incorrect:

Figure 15-98. Timing Chart of Master Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)



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

Figure 15-98. Timing Chart of Master Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)



No.23: Change the register setting order in the figure in the same way as in the Figure 15-101.

Page: P.934

Incorrect:

Figure 15-100. Example of Contents of Registers for Slave Transmission of SPI Function (CSI00, CSI01, CSI10, CSI11) (1/2)

(d) Serial slave select enable register m (SSEm) … Controls the SSI00, SSI01, SSI10, and SSI11 pin inputs of the target channel in slave 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

(e) Serial output register m (SOm) … Sets only the bits of the target channel.

	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

(f) 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
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	SOEm0 0/1

(g) Serial channel start register m (SSm) … 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
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1

Correct:

Figure 15-100. Example of Contents of Registers for Slave Transmission of SPI Function (CSI00, CSI01, CSI10, CSI11) (2/2)

(d) Serial output register m (SOm) … Sets only the bits of the target channel.

	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
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	SOEm0 0/1

(f) Serial slave select enable register m (SSEm) … Controls the SSI00, SSI01, SSI10, and SSI11 pin inputs of the target channel in slave 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

(g) Serial channel start register m (SSm) … 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
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1

No.24: Correct the content in “Remark 1” below the figure.

Page: P.936

Incorrect:

Figure 15-102. Procedure for Stopping Slave Transmission

Remarks 1. Even after communication is stopped, the pin level is retained. To resume the operation, re-set serial output register m (SOm) (see **Figure 15-103 Procedure for Resuming Slave Transmission**).

Correct:

Figure 15-102. Procedure for Stopping Slave Transmission

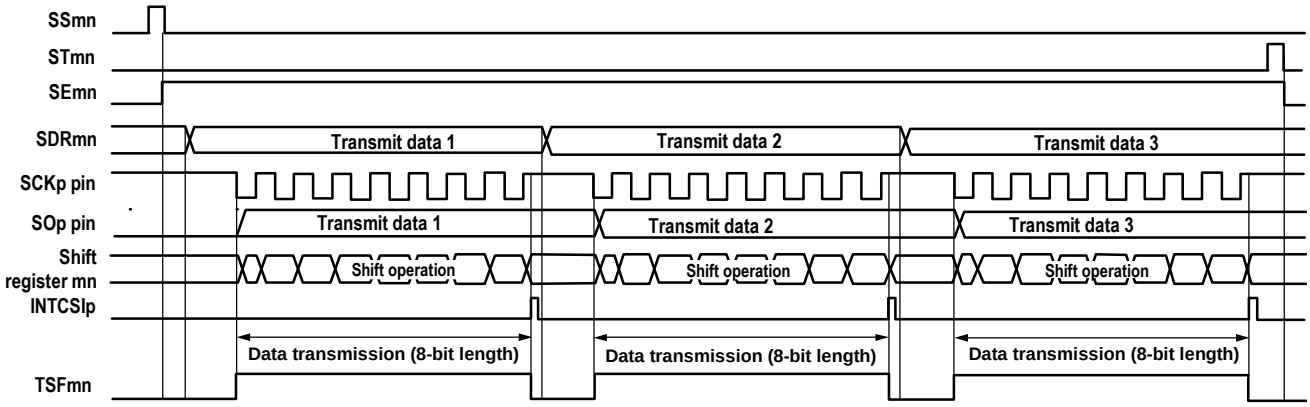
Remarks 1. When the SSImn input signal is low, the pin level is retained even after communication is stopped. To resume the operation, re-set SOm register (see **Figure 15-103 Procedure for Resuming Slave Transmission**).

No.25: Add the SSIp pin in the figure. Also correct the output of the SOp pin.

Page: P.938

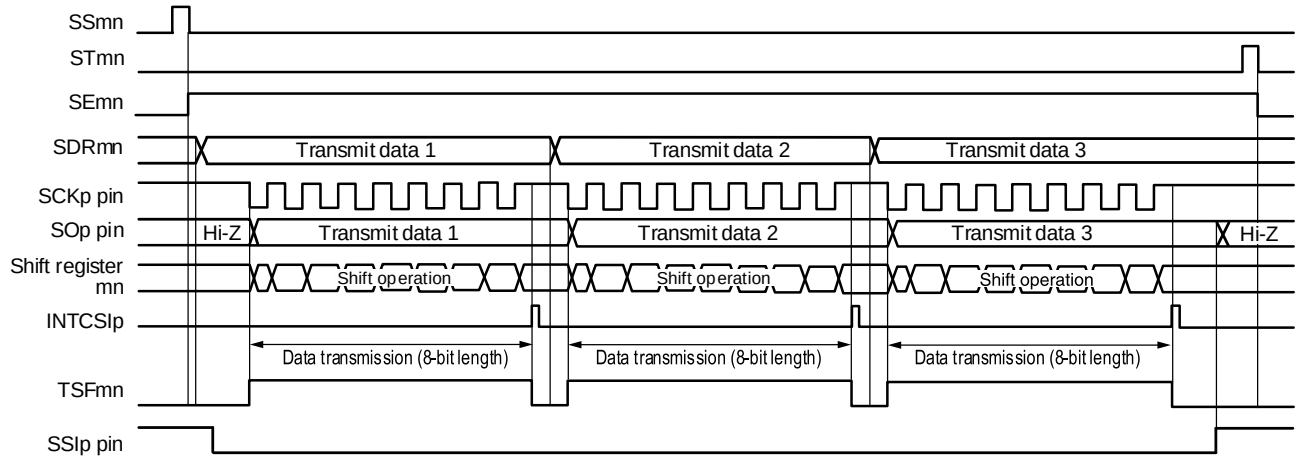
Incorrect:

Figure 15-104. Timing Chart of Slave Transmission (in Single-Transmission Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



Correct: For slave 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)

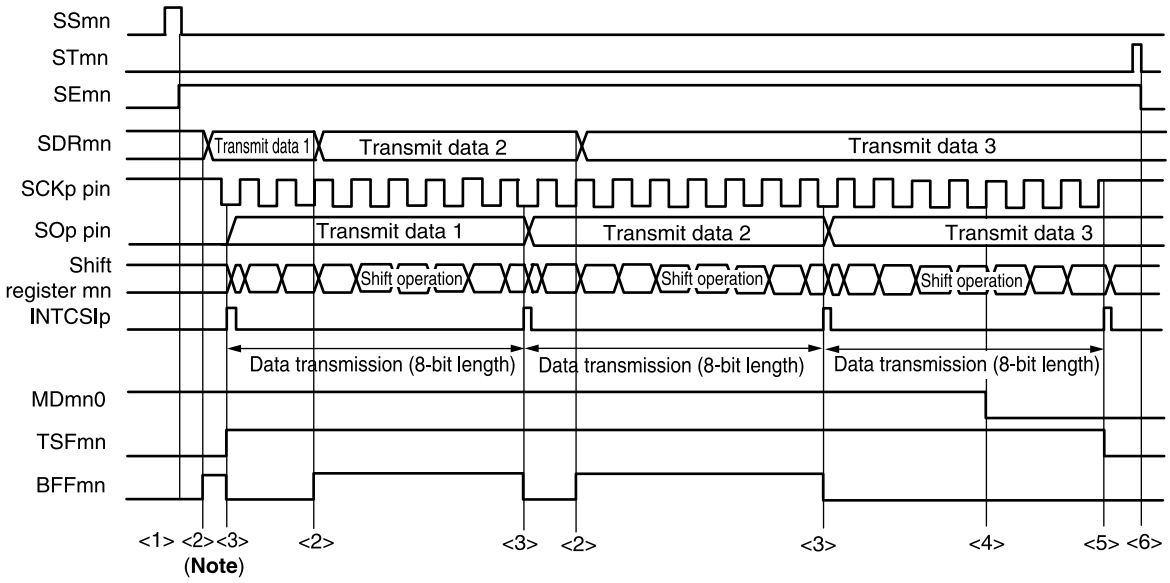


No.26: Add the SSIp pin in the figure. Also correct the output of the SOp pin.

Page: P.940

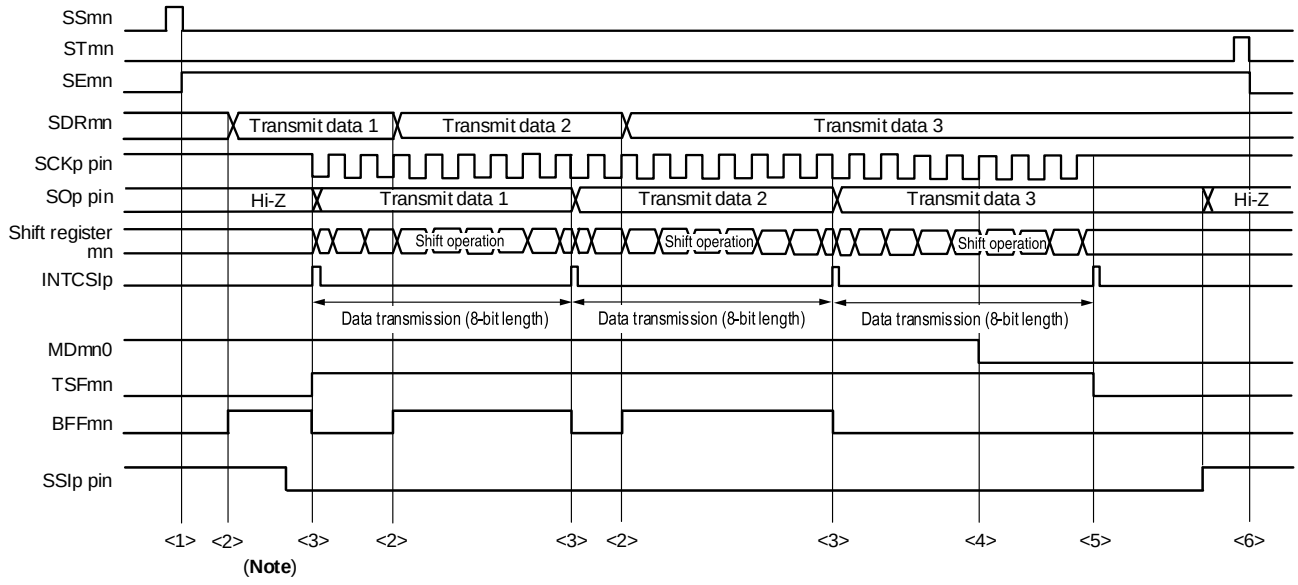
Incorrect:

Figure 15-106. Timing Chart of Slave Transmission (in Continuous Transmission Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



Correct: For slave transmission, add “SSIp pin signal” to the figure.

Figure 15-106. Timing Chart of Slave Transmission (in Continuous Transmission Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



No.27: Change the register setting order in the figure in the same way as in the Figure 15-109.

Page: P.944

Incorrect:

Figure 15-108. Example of Contents of Registers for Slave Reception of SPI Function (CSI00, CSI01, CSI10, CSI11) (2/2)

(e) Serial output enable register m (SOEm) …The register that not used in this mode.

	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) Serial channel start register m (SSm) … 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
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1

(g) Serial slave select enable register m (SSEm) … Controls the SSI00, SSI01, SSI10, and SSI11 pin inputs of the target channel in slave 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

Correct:

Figure 15-108. Example of Contents of Registers for Slave Reception of SPI Function (CSI00, CSI01, CSI10, CSI11) (2/2)

(e) Serial output enable register m (SOEm) …The register that not used in this mode.

	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) Serial slave select enable register m (SSEm) … Controls the SSI00, SSI01, SSI10, and SSI11 pin inputs of the target channel in slave 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

(g) Serial channel start register m (SSm) … Sets only the bits of the target channel to 1.

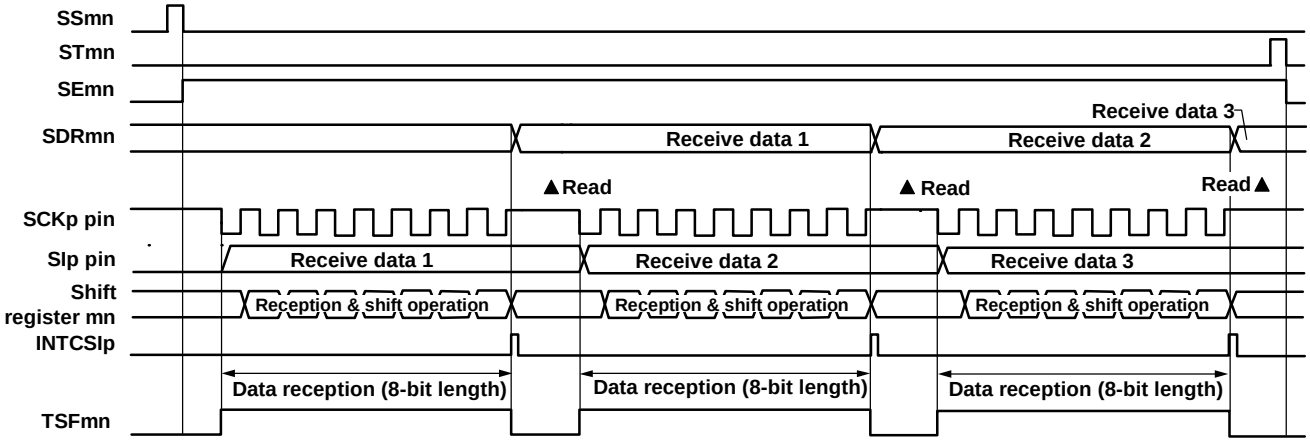
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1

No.28: Add the SSIp pin in the figure.

Page: P.947

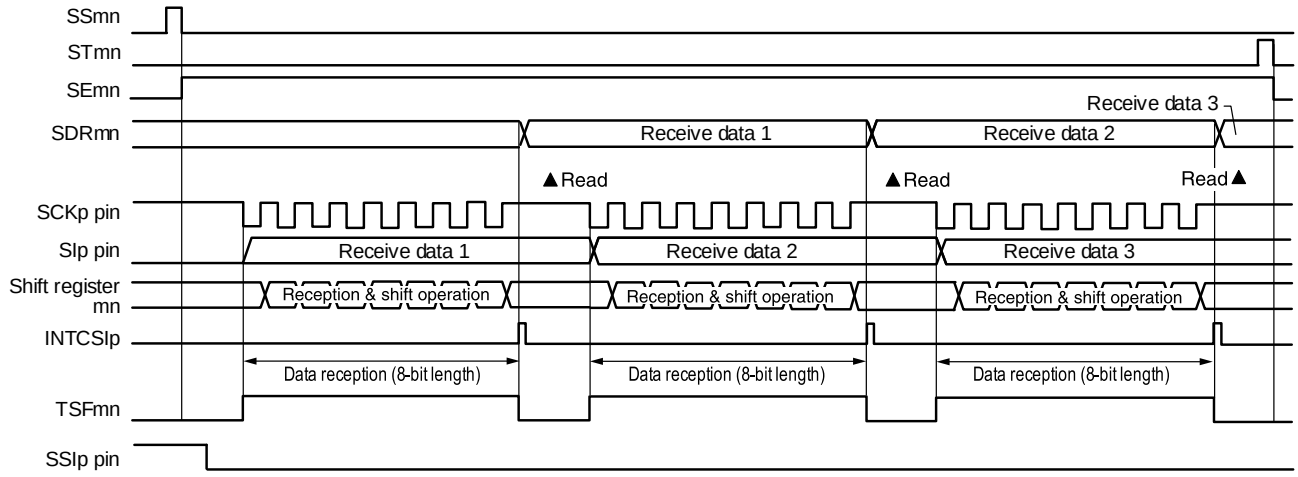
Incorrect:

Figure 15-112. Timing Chart of Slave Reception (in Single-Reception Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



Correct: For slave reception, add “SSIp pin signal” to the figure.

Figure 15-112. Timing Chart of Slave Reception (in Single-Reception Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



No.29: Change the register setting order in the figure in the same way as in the Figure 15-115.

Page: P.951

Incorrect:

Figure 15-114. Example of Contents of Registers for Slave Transmission/Reception of SPI Function (CSI00, CSI01, CSI10, CSI11) (2/2)

(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
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	SOEm0 0/1

(f) Serial channel start register m (SSm) … 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
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1

(g) Serial slave select enable register m (SSEm) … Controls the SSI00, SSI01, SSI10, and SSI11 pin inputs of the target channel in slave 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

Correct:

Figure 15-114. Example of Contents of Registers for Slave Transmission/Reception of SPI Function (CSI00, CSI01, CSI10, CSI11) (2/2)

(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
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm1 0/1	SOEm0 0/1

(f) Serial slave select enable register m (SSEm) … Controls the SSI00, SSI01, SSI10, and SSI11 pin inputs of the target channel in slave 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

(g) Serial channel start register m (SSm) … Sets only the bits of the target channel to 1.

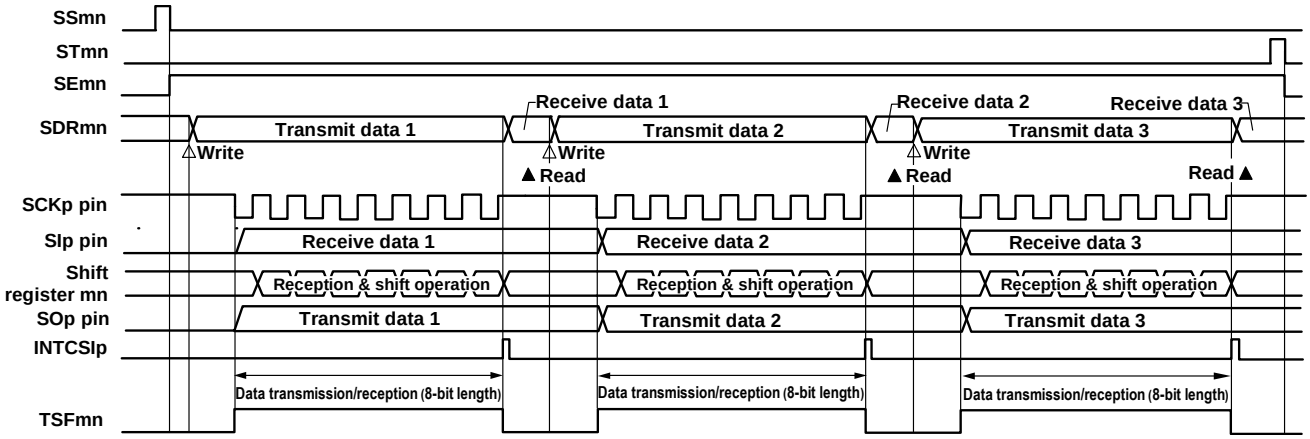
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSm1 0/1	SSm0 0/1

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

Page: P.955

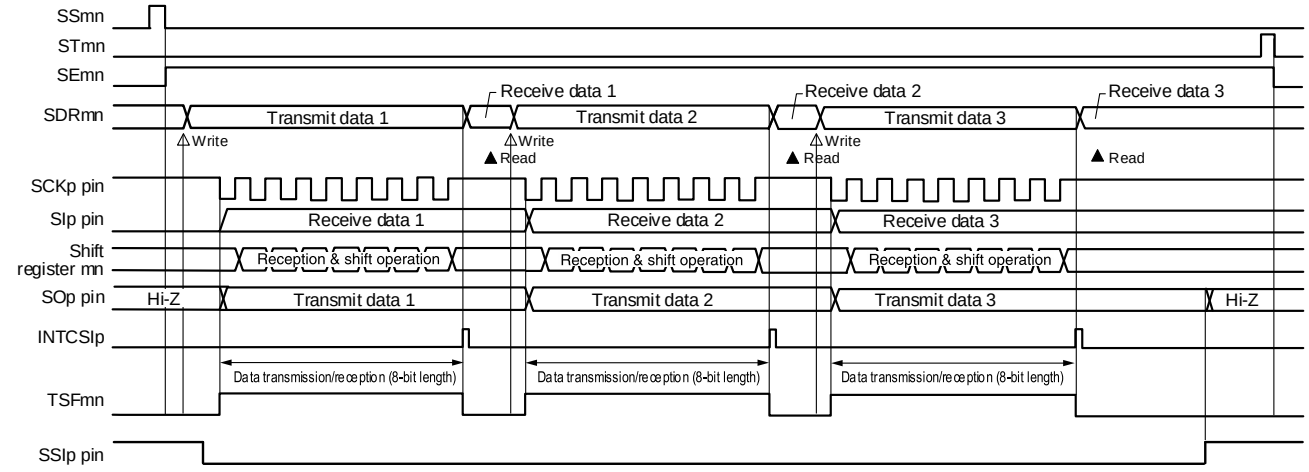
Incorrect:

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)

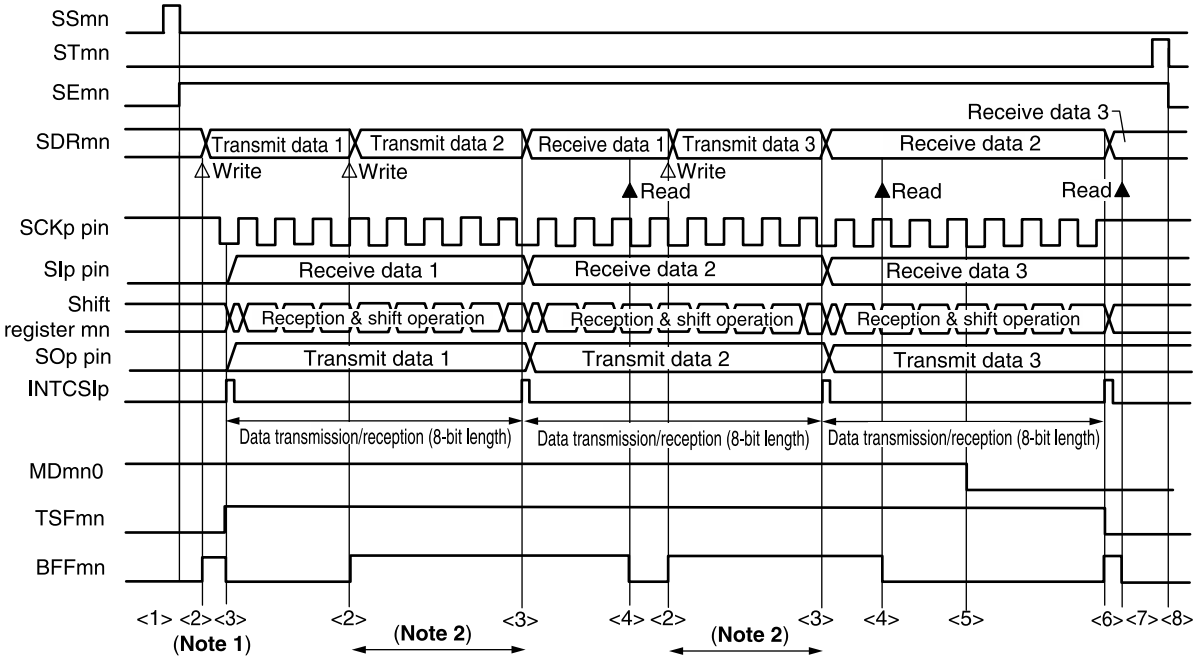


No.31: Add the SSIp pin in the figure. Also correct the output of the SOp pin.

Page: P.957

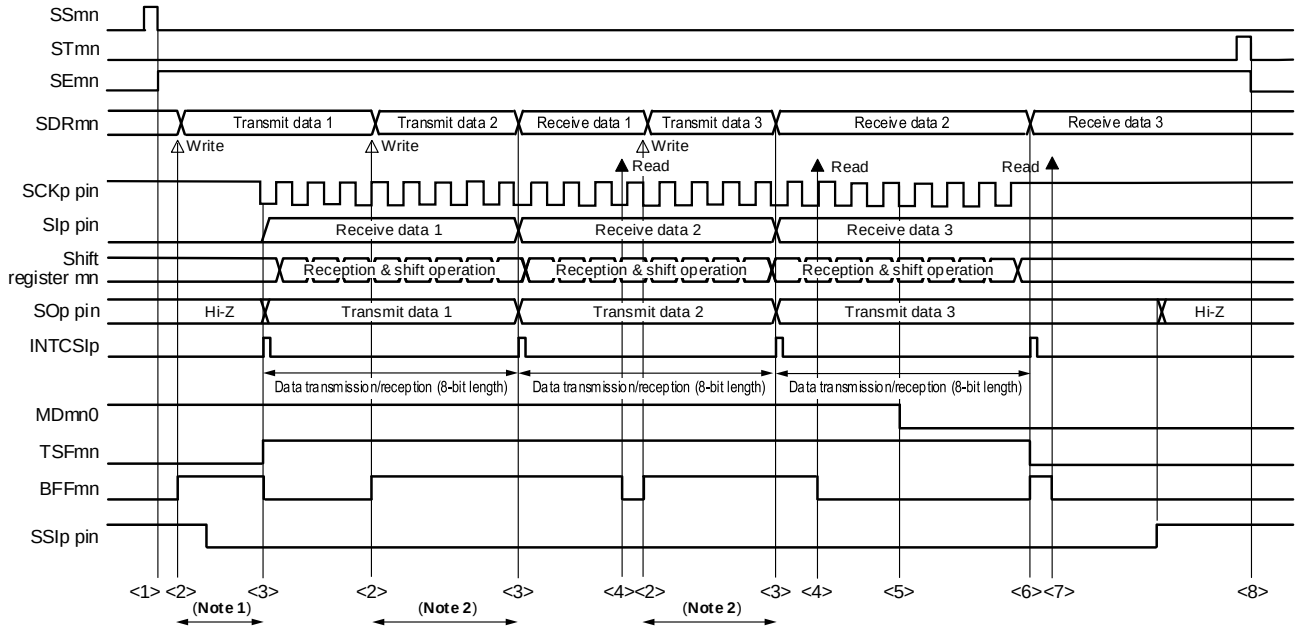
Incorrect:

Figure 15-120. Timing Chart of Slave Transmission/Reception (in Continuous 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 15-120. Timing Chart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)



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)

Address: F06CEH

Symbol	7	6	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

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”.

No.35: Correct typos in the “Caution” below the table.

Page: P.1143

Incorrect:

17.2.2 LIN Registers for Slave Mode

(3) Peripheral Enable Register 2 (PER2)

Address: F02C1H After reset: 00H R/W

Symbol	7	<6>	5	4	<3>	<2>	1	<0>
PER2	0	AAUEN	0	0	LIN1EN Note	LIN0EN	0	CAN0EN Note

Caution Be sure to clear the following bits to 0. Bits 0, 3, 4, 5, and 7 in the RL78/F23 products. Bits 4, 5, and 7 in the RL78/F24 products.

Correct:

17.2.2 LIN Registers for Slave Mode

(3) Peripheral Enable Register 2 (PER2)

Address: F02C1H After reset: 00H R/W

Symbol	7	<6>	5	4	<3>	<2>	1	<0>
PER2	0	AAUEN	0	0	LIN1EN Note	LIN0EN	0	CAN0EN Note

Caution Be sure to clear the following bits to 0. Bits 0, 1, 3, 4, 5, and 7 in the RL78/F23 products. Bits 1, 4, 5, and 7 in the RL78/F24 products.

No.36: Add “Caution” regarding LIN reset mode transitions.

Page: P.1156

Correct:

17.2.2 LIN Registers for Slave Mode

(15) LIN/UART Control Register (LCUCn)

Address: F06CEH

Symbol	7	6	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

Caution When transitioning from LIN mode with timeout error detection enabled (TERE 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”.

No.37: Correct typos in the “Caution” below the table.

Page: P.1172

Incorrect:

17.2.3 Registers for UART

(3) Peripheral Enable Register 2 (PER2)

Address: F02C1H After reset: 00H R/W

Symbol	7	<6>	5	4	<3>	<2>	1	<0>
PER2	0	AAUEN	0	0	LIN1EN Note	LIN0EN	0	CAN0EN Note

Caution Be sure to clear the following bits to 0. Bits 0, 3, 4, 5, and 7 in the RL78/F23 products. Bits 4, 5, and 7 in the RL78/F24 products.

Correct:

17.2.3 Registers for UART

(3) Peripheral Enable Register 2 (PER2)

Address: F02C1H After reset: 00H R/W

Symbol	7	<6>	5	4	<3>	<2>	1	<0>
PER2	0	AAUEN	0	0	LIN1EN Note	LIN0EN	0	CAN0EN Note

Caution Be sure to clear the following bits to 0. Bits 0, 1, 3, 4, 5, and 7 in the RL78/F23 products. Bits 1, 4, 5, and 7 in the RL78/F24 products.

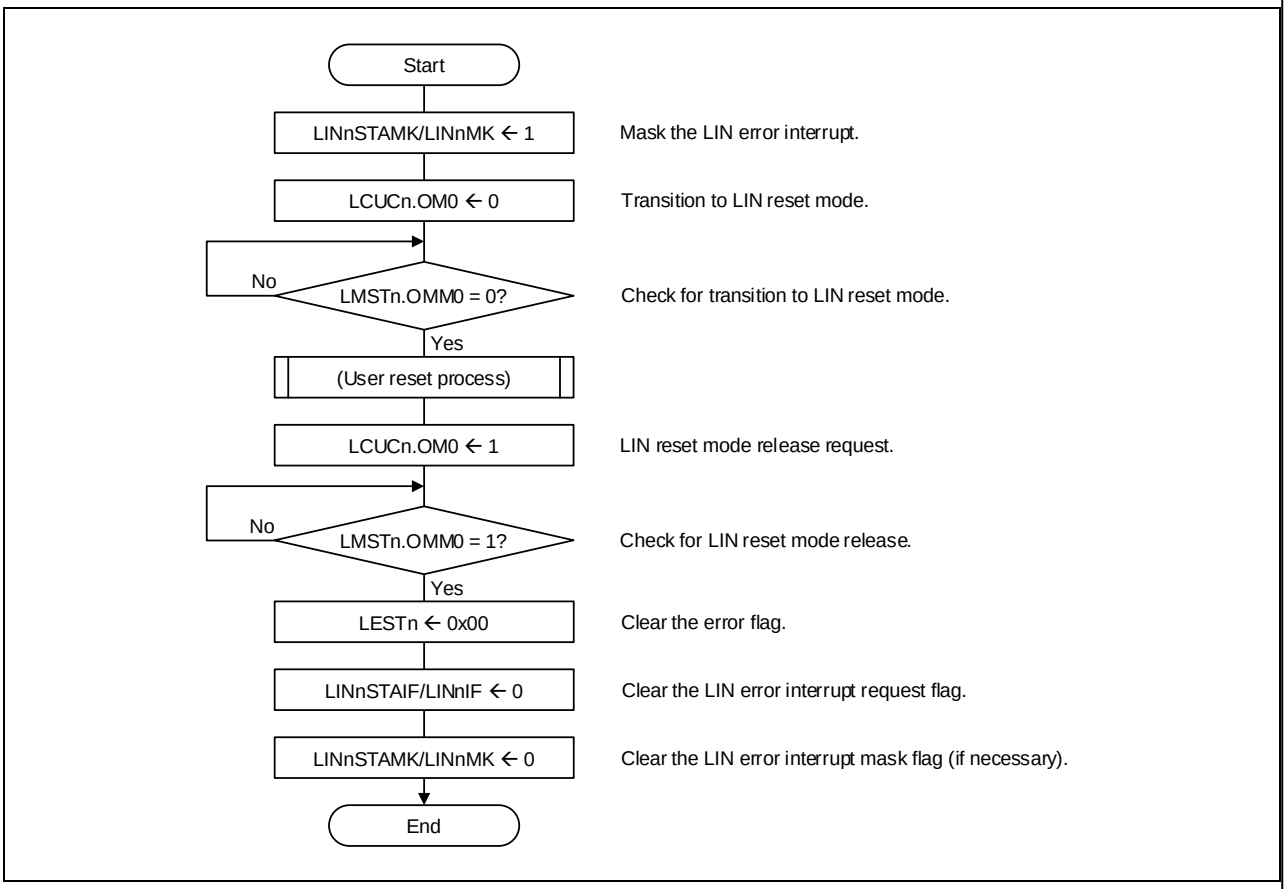
No.38: Add the “Caution” about transitioning to LIN reset mode when timeout error detection is enabled.

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.



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	–	–	–	–	DSJW[3:0]				–	–	–	–	DTSEG2[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
C0DCFGL	–	–	–	DTSEG1[4:0]					DBRP[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 f_{CAN} by P + 1.
  The CAN0 Tq clock (f_{CANTQ0}) is obtained by the CAN communication clock (f_{CAN}) 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	–	–	–	–	DSJW[3:0]				–	–	–	–	DTSEG2[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
C0DCFGL	–	–	–	DTSEG1[4:0]					DBRP[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_{CANTQ0}) is obtained by the CAN communication clock (f_{CAN}) 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. 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 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.

No.43: Correct the typo in the reserved word definition for TMIF12 bit.

Page: P.1561

Incorrect:

Figure 21-2. Format of Interrupt Request Flag Registers (IFxL, IFxH) (2/2)

Address: FFFD1H After reset: 00H R/W

Symbol	7	<6>	**5**	<4>	<3>	<2>	<1>	<0>
IF2H	0	TMIF13	TMIF12	TMIF11	TMIF10	RCANGERRIF	RCANGREFRIF	RCAN0TRMIF

Correct:

Figure 21-2. Format of Interrupt Request Flag Registers (IFxL, IFxH) (2/2)

Address: FFFD1H After reset: 00H R/W

Symbol	7	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF2H	0	TMIF13	TMIF12	TMIF11	TMIF10	RCANGERRIF	RCANGREFRIF	RCAN0TRMIF

No.44: Delete unnecessary references in “Note 4” below the figure.

Page: P.1569

Incorrect:

Figure 21-6. Format of Interrupt Source Determination Flag Register 0 (INTFLG0)

Address: F0079H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
INTFLG0	INTFLG07 Notes 3, 5	INTFLG06 Notes 2, 4	0	0	0	INTFLG02 Notes 1, 5	INTFLG01 Note 5	INTFLG00 Note 5

Note 4. Even if the RPTINT bit in the DTCCRj register (j = 0 to 23) is set to 0 (disabling the interrupt while the DTC module is in repeat mode), when the comparator detection 0 interrupt source is generated, the INTFLG06 bit is set to 1. **For details, see (A) Internal maskable interrupt (only comparator detection 0 interrupt) in Figure 21-1. Basic Configuration of Interrupt Function.**

Correct:

Figure 21-6. Format of Interrupt Source Determination Flag Register 0 (INTFLG0)

Address: F0079H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
INTFLG0	INTFLG07 Notes 3, 5	INTFLG06 Notes 2, 4	0	0	0	INTFLG02 Notes 1, 5	INTFLG01 Note 5	INTFLG00 Note 5

Note 4. Even if the RPTINT bit in the DTCCRj register (j = 0 to 23) is set to 0 (disabling the interrupt while the DTC module is in repeat mode), when the comparator detection 0 interrupt source is generated, the INTFLG06 bit is set to 1.

No.45: Correct typos in the reserved word definition for ISC0, ISC2, and ISC3 bits.

Page: P.1573

Incorrect:

Figure 21-9. Format of Input Switch Control Register (ISC)

Address: F0073H After reset: 00H R/W

Symbol	7	6	5	4	<3>	<2>	1	<0>
ISC	0	0	0	0	ISC3	ISC2	0	ISC0

Correct:

Figure 21-9. Format of Input Switch Control Register (ISC)

Address: F0073H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ISC	0	0	0	0	ISC3	ISC2	0	ISC0

No.46: Add an example of the state when interrupt request hold instructions are consecutive.

Page: P.1584

Correct:

21.4.5 Interrupt Request Hold

Note If the interrupt request hold instructions are consecutive, the interrupt hold state will also continue.

Example 1) Interrupt requests are held pending.

```
EI ; Enable interrupt instruction      Interrupt request hold instruction
DI ; Disable interrupt instruction     Interrupt request hold instruction
```

Example 2) Interrupt request is accepted.

```
EI ; Enable interrupt instruction      Interrupt request hold instruction
NOP ;
DI ; Disable interrupt instruction     Interrupt request hold instruction
```

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 Note 1	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 Note 2	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.

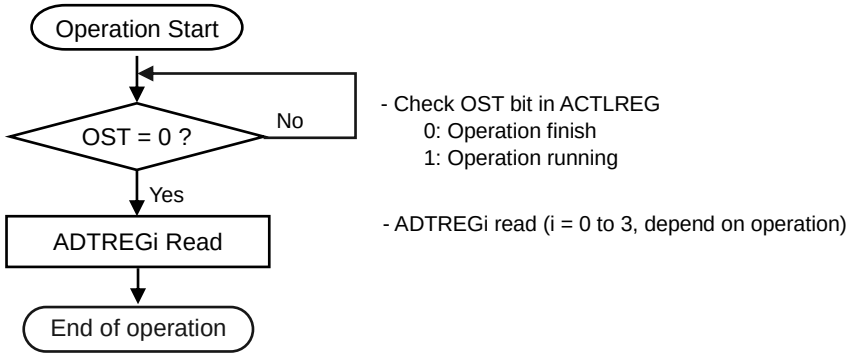
No.50: Add “NOP instruction (1 cycle)” in the figure.

Page: P.1671

Incorrect:

Figure 27-37. Processing Flow of Read of Operation Results

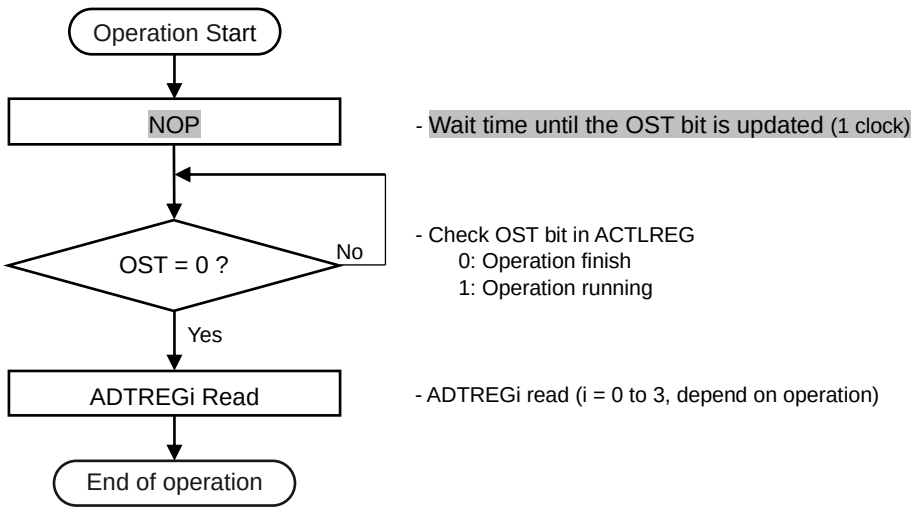
(a) Operation status check case



Correct:

Figure 27-37. Processing Flow of Read of Operation Results

(a) Operation status check case



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 */
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;
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 */
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: 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);
```

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 */
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 – AL1OFS);
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 – AL1OFS);
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;
```

No.54: Correct a typo in the ERADR register.

Page: P.1707

Incorrect:

28.3.3 Internal RAM-ECC Function

(1) Error address store register (ERADR)

Figure 28-8. Format of Error Address Store Register (ERADR)

Address: F0200H After reset: 0000H **R/W**

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

Cautions 1. The ERADR register can be set by a 16-bit memory manipulation instruction.

2. The register value is updated each time a bit error interrupt request is generated.

Correct:

28.3.3 Internal RAM-ECC Function

(1) Error address store register (ERADR)

Figure 28-8. Format of Error Address Store Register (ERADR)

Address: F0200H After reset: 0000H R

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

Cautions 1. The ERADR register should be read by a 16-bit memory manipulation instruction.

2. The register value is updated each time a bit error interrupt request is generated.

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]	**RRADRnH**.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]

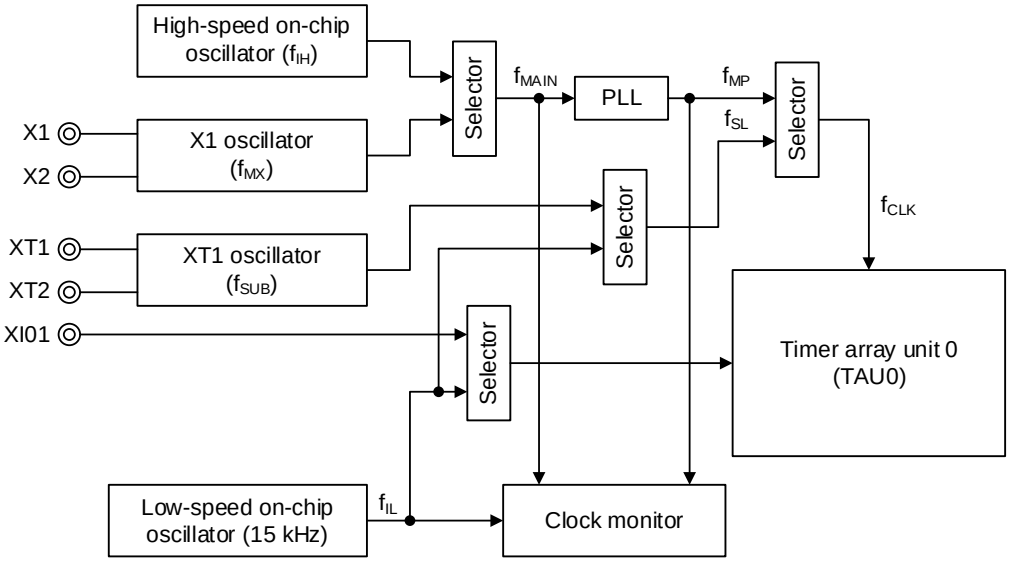
No.56: Correct typo in pin function name.

Page: P.1749

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

28.3.9 Frequency Detection Function

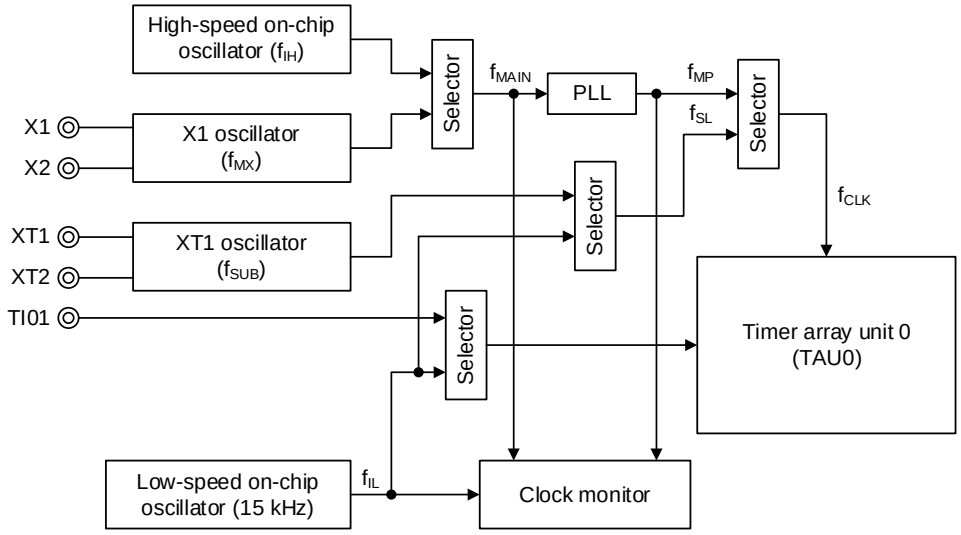
Figure 28-43. Configuration of Frequency Detection Function



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

28.3.9 Frequency Detection Function

Figure 28-43. Configuration of 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)

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

OCDENSET	HPIEN Note 3	OCDERSD	Control of on-chip debug operation
0	0	0	Disables on-chip debug operation.
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-chip debugging and disables hot plug-in operation. Does not erase data of flash memory in case of failures in authenticating on-chip debug security ID.
1	1	1	Enables on-chip debugging and hot plug-in operation. Does not erase data of flash memory in case of failures in authenticating on-chip debug security ID.
Other than the above			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	X	X	X	Disables flash serial programming and on-chip debugging operation.
1	0	0	0	Disables on-chip debugging.
1	1	0	0	• Enables on-chip debugging. • Disables hot plug-in operation. • Erases data of flash memory in case of failures in authenticating on-chip debug security ID.
1	1	0	1	• Enables on-chip debugging. • Disables hot plug-in operation. • Does not erase data of flash memory in case of failures in authenticating on-chip debug security ID.
1	1	1	1	• Enables on-chip debugging. • Enables hot plug-in operation. • Does not erase data of flash memory in case of failures in authenticating on-chip debug security ID.
1	Other than the above			Setting prohibited

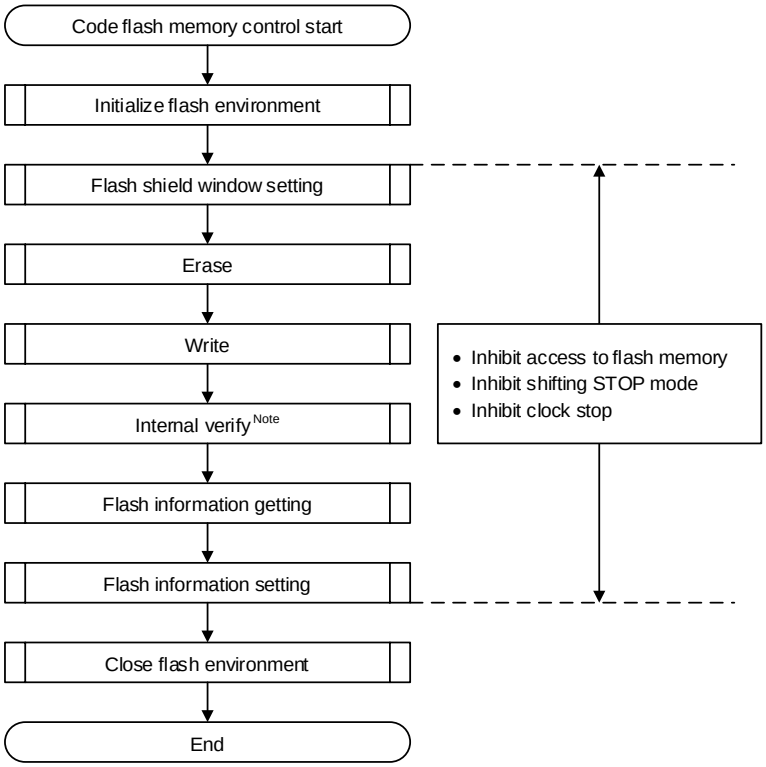
No.58: Add “HALT mode to prohibited transitions” to flow annotations.

Page: P.1778

Incorrect: Inhibit setting STOP mode

32.7.1 Self-Programming Procedure

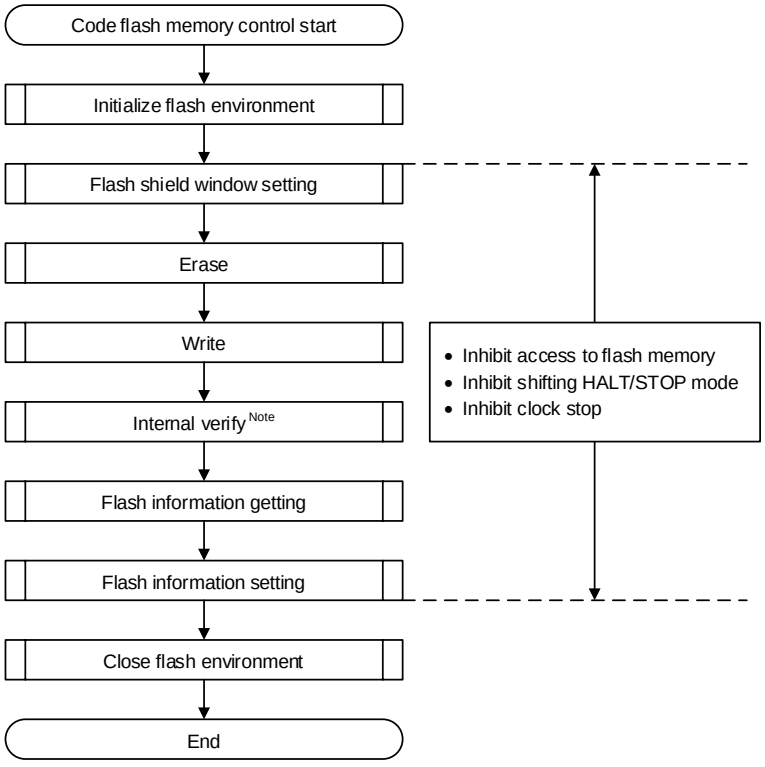
Figure 32-8. Flow of Self-Programming (Rewriting Flash Memory)



Correct: Inhibit setting HALT/STOP mode

32.7.1 Self-Programming Procedure

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



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:

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

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Bit manipulate	OR1	CY, A.bit	2	1	–	CY ← CY ∨ A.bit			×
		CY, PSW.bit	3	1	–	**CYX ← CY ∨ PSW.bit**			×
		CY, saddr.bit	3	1	–	CY ← CY ∨ (saddr).bit			×
		CY, sfr.bit	3	1	–	CY ← CY ∨ sfr.bit			×
		CY, [HL].bit	2	1	4	CY ← CY ∨ (HL).bit			×
		CY, ES:[HL].bit	3	2	5	CY ← CY ∨ (ES, HL).bit			×

Correct:

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

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Bit manipulate	OR1	CY, A.bit	2	1	–	CY ← CY ∨ A.bit			×
		CY, PSW.bit	3	1	–	CY ← CY ∨ PSW.bit			×
		CY, saddr.bit	3	1	–	CY ← CY ∨ (saddr).bit			×
		CY, sfr.bit	3	1	–	CY ← CY ∨ sfr.bit			×
		CY, [HL].bit	2	1	4	CY ← CY ∨ (HL).bit			×
		CY, ES:[HL].bit	3	2	5	CY ← CY ∨ (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 (T_A) 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		f_{CLK} = 2 MHz		f_{CLK} = 4 MHz		f_{CLK} = 8 MHz		f_{CLK} = 16 MHz		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	–	19.0	–	9.5	–	5.0	–	2.5	ms

Item		f_{CLK} = 20 MHz		f_{CLK} = 32 MHz		f_{CLK} = 40 MHz		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		f_{CLK} = 2 MHz		f_{CLK} = 4 MHz		f_{CLK} = 8 MHz		f_{CLK} = 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	–	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		f_{CLK} = 20 MHz		f_{CLK} = 32 MHz		f_{CLK} = 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.

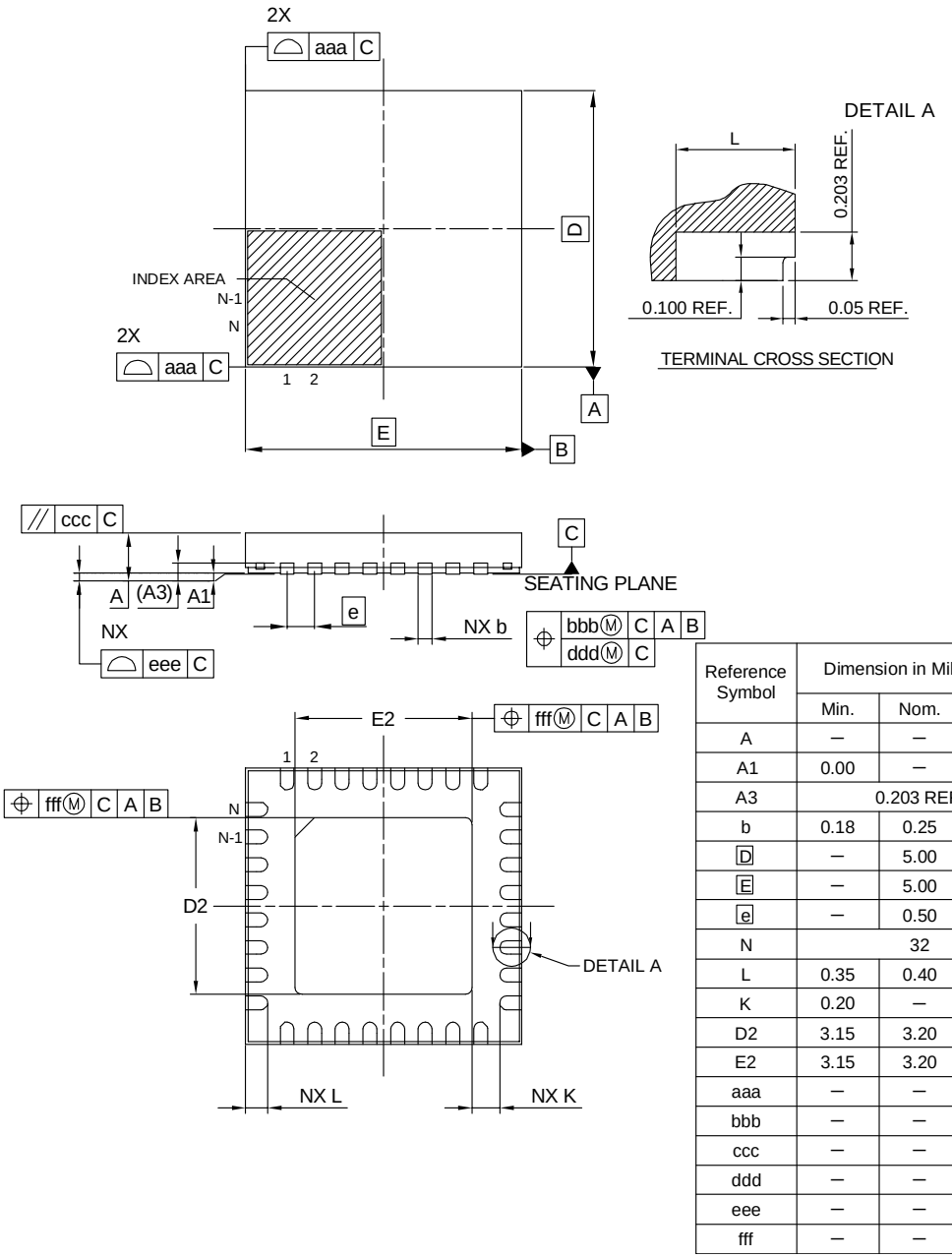
No.69: Correct the value of the pin cross section in the figure.

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

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN032-5x5-0.50	PWQN0032KF-B	0.06

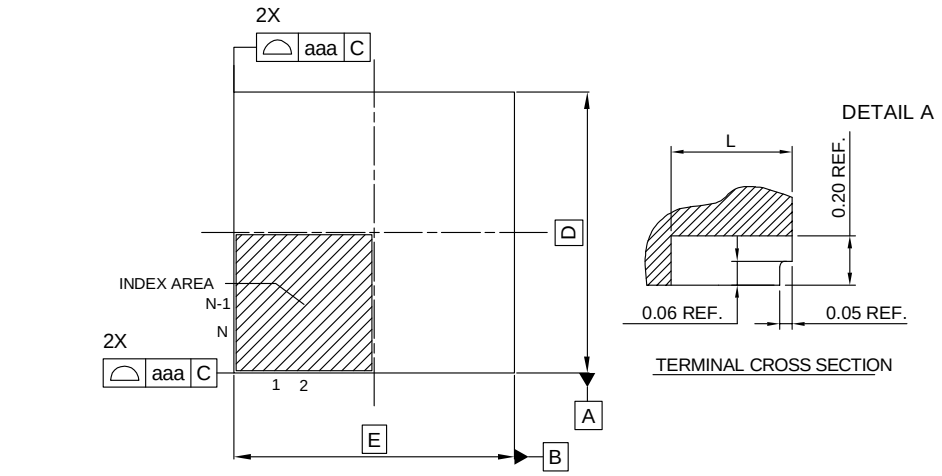


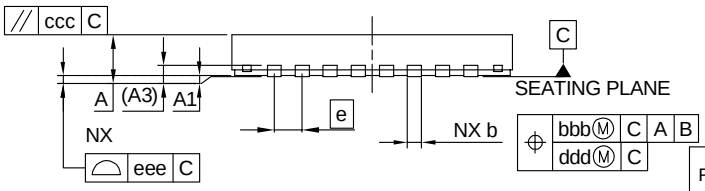
Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
A	—	—	0.80
A1	0.00	—	0.05
A3		0.203 REF.	
b	0.18	0.25	0.30
D	—	5.00	—
E	—	5.00	—
e	—	0.50	—
N		32	
L	0.35	0.40	0.45
K	0.20	—	—
D2	3.15	3.20	3.25
E2	3.15	3.20	3.25
aaa	—	—	0.15
bbb	—	—	0.10
ccc	—	—	0.10
ddd	—	—	0.05
eee	—	—	0.08
fff	—	—	0.10

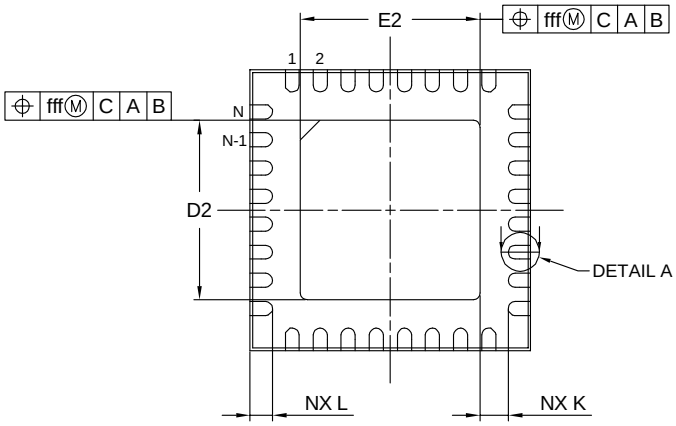
Correct: Terminal cross section: [0.06 REF, 0.05 REF, 0.20 REF], A3: 0.20 REF

39.1 32-pin products

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN032-5x5-0.50	PWQN0032KF-B	0.06







Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
A	–	–	0.80
A1	0.00	–	0.05
A3		0.20 REF.	
b	0.18	0.25	0.30
D	–	5.00	–
E	–	5.00	–
e	–	0.50	–
N		32	
L	0.35	0.40	0.45
K	0.20	–	–
D2	3.15	3.20	3.25
E2	3.15	3.20	3.25
aaa	–	–	0.15
bbb	–	–	0.10
ccc	–	–	0.10
ddd	–	–	0.05
eee	–	–	0.08
fff	–	–	0.10