

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-RA*-A0159A/E	Rev.	1.00
Title	RA0E1 Description in the User's Manual: Hardware Rev.1.10 Changed.		Information Category	Technical Notification	
Applicable Product	RA0E1 Group	Lot No.	Reference Document	RA0E1 Group User's Manual: Hardware Rev.1.10 (R01UH1040EJ0110)	
		All			

This document describes misstatements found in the RA0E1 User's Manual: Hardware Rev. 1.10 (R01UH1040EJ0110).

Corrections

Applicable Item	Applicable Page	Contents	Pages in this document for corrections
Table 1.15 Pin list	Page 39	Incorrect descriptions revised	Page 2, 3
9.2.1 SBYCR : Standby Control Register	Page 103	Incorrect descriptions revised	Page 4
Table 16.6 Examples of register settings for port and alternate functions (2/6) (2 of 3)	Page 196	Incorrect descriptions revised	Page 5, 6
Table 16.6 Examples of register settings for port and alternate functions (2/6) (3 of 3)	Page 197	Incorrect descriptions revised	Page 7
16.6 Peripheral Select Settings for Each Product	Page 200	Incorrect descriptions revised	Page 8
22.2.7 IICWL0 : IICA Low-level Width Setting Register 0	Page 475	Caution added	Page 9
22.2.8 IICWH0 : IICA High-level Width Setting Register 0	Page 476	Caution added	Page 10
22.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers	Page 477	Caution added	Page 11, 12
22.3.10 Timing of Generation of the Interrupt Request Signal (IICAn_TXRXI) and Control of Clock Stretching	Page 482, 483	Incorrect descriptions revised	Page 13, 14
23.3.4 Baud Rate Generator	Page 548, 549	Incorrect descriptions revised	Page 15, 16
28.3.25 PNRn : Part Numbering Register n (n = 0 to 3)	Page 636	Incorrect descriptions revised	Page 17, 18
Table 31.12 Operating and standby current (1) (2 of 2)	Page 672	Caution added	Page 19, 20

Document Improvement

The above corrections will be made for the next revision of the User's Manual: Hardware.

[Before correction of Table 1.15 Pin list] (Page 39)

1.7 Pin Lists

Table 1.15 Pin list

Pin number				Power, System, Clock, Debug	I/O ports	Interrupt	Timers		Communication interfaces			Analogs
32-pin	24-pin	20-pin	16-pin				TAU	RTC	SAU	IICA	UARTA	ADC
1	1	2	16	VCL	—	—	—	—	—	—	—	—
2	—	—	—	XCIN	P215	—	—	—	—	—	—	—
3	—	—	—	XCOUT	P214	—	—	—	—	—	—	—
4	2	3	1	VSS	—	—	—	—	—	—	—	—
5	3	4	2	X2/EXCLK/ XCOUT ^{*1}	P213	IRQ0_B	T100_A/T102_B/ TO02_B	—	TXD1_A/ SO11_A	SDAA0_B	TXDA0_B	—
6	4	5	3	X1/XCIN ^{*1}	P212	IRQ1_B	TO00_A/ T103_C/TO03_C	—	RXD1_A/ SI11_A/ SDA11_A	SCLA0_B	RXDA0_B	—
7	5	6	4	VCC	—	—	—	—	—	—	—	—
8	—	—	—	PCLBUZ0_C	P407	IRQ4_C	—	RTCCOUT_A	SCK11_A/ SCL11_A	—	—	—
9	6	—	—	—	P914	—	—	—	—	SCLA0_A	—	—
10	7	—	—	—	P913	—	—	—	—	SDAA0_A	—	—
11	—	—	—	—	P208	IRQ3_C	T100_B	—	—	—	TXDA0_A	—
12	—	—	—	—	P207	IRQ2_C	TO00_B	—	—	—	RXDA0_A	—
13	8	7	5	RES	P206	—	—	—	—	—	—	—
14	9	8	6	PCLBUZ0_A	P201	IRQ5_B	T105_B/TO05_B	RTCCOUT_B	SSI00_B/ SCK11_B/ SCL11_B	—	—	—
15	10	9	7	—	P200	IRQ0_A/NMI	—	—	—	—	—	—
16	11	10	8	SWCLK	P300	—	T104_B/TO04_B	—	—	—	—	—
17	12	11	9	SWDIO	P108	—	T103_B/TO03_B	—	—	—	—	—
18	13	12	—	—	P109	IRQ4_B	T102_A/TO02_A	—	TXD2_A/ SO20_A	SDAA0_C	TXDA0_C	—
19	14	13	—	—	P110	IRQ3_B	T101_A/TO01_A	—	RXD2_A/ SI20_A/ SDA20_A	SCLA0_C	RXDA0_C	—
20	15	14	—	—	P112	IRQ2_B	T103_A/TO03_A	—	SCK20_A/ SCL20_A/ SSI00_C	—	—	—
21	—	—	—	—	P103	IRQ5_A	T105_A/TO05_A	—	SSI00_A	—	—	—
22	16	15	10	PCLBUZ0_B	P102	IRQ4_A	T106_A/ TO06_A/ TO00_C	RTCCOUT_C	SCK00_A/ SCL00_A	—	—	—
23	17	16	11	—	P101	IRQ3_A	T107_A/ TO07_A/T100_C	—	TXD0_A/ SO00_A	SDAA0_D	TXDA0_D	AN021
24	18	17	12	—	P100	IRQ2_A	T104_A/ TO04_A/ T101_B/TO01_B	—	RXD0_A/ SI00_A/ SDA00_A	SCLA0_D	RXDA0_D	AN022
25	19	—	—	—	P015	IRQ1_A	—	—	—	—	—	AN007
26	20	—	—	—	P014	—	—	—	—	—	—	AN006
27	21	18	—	—	P013	—	—	—	—	—	—	AN005
28	22	19	13	—	P012	—	—	—	—	—	—	AN004
29	—	—	—	—	P009	—	—	—	—	—	—	AN003
30	—	—	—	—	P008	—	—	—	—	—	—	AN002
31	23	20	14	VREFL0	P011	—	—	—	—	—	—	AN001
32	24	1	15	VREFH0	P010	—	—	—	—	—	—	AN000

Note 1. When setting CMC.XTSEL = 1 for 24-, 20-, and 16-pin products

Note: Some signal names have _A, _B, _C or _D suffixes, but these suffixes can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for SCL11 and SCK11, bearing the same suffix can be selected. The simultaneous use of the same signal with different suffixes is prohibited.

[After correction of Table 1.15 Pin list]

1.7 Pin Lists

Table 1.15 Pin list

Pin number				Power, System, Clock, Debug	I/O ports	Interrupt	Timers		Communication interfaces			Analogs
32-pin	24-pin	20-pin	16-pin				TAU	RTC	SAU	IICA	UARTA	ADC
1	1	2	16	VCL	—	—	—	—	—	—	—	—
2	—	—	—	XCIN	P215	—	—	—	—	—	—	—
3	—	—	—	XCOUT	P214	—	—	—	—	—	—	—
4	2	3	1	VSS	—	—	—	—	—	—	—	—
5	3	4	2	X2/EXCLK/ XCOUT ^{*1}	P213	IRQ0_B	T100_A/T102_B/ TO02_B	—	TXD1_A/ SO11_A	SDAA0_B	TXDA0_B	—
6	4	5	3	X1/XCIN ^{*1}	P212	IRQ1_B	TO00_A/ TI03_C/TO03_C	—	RXD1_A/ SI11_A/ SDA11_A	SCLA0_B	RXDA0_B	—
7	5	6	4	VCC	—	—	—	—	—	—	—	—
8	—	—	—	PCLBUZ0_C	P407	IRQ4_C	—	RTCCOUT_A	SCK11_A/ SCL11_A	—	—	—
9	6	—	—	—	P914	—	—	—	—	SCLA0_A	—	—
10	7	—	—	—	P913	—	—	—	—	SDAA0_A	—	—
11	—	—	—	—	P208	IRQ3_C	TI00_B	—	—	—	TXDA0_A	—
12	—	—	—	—	P207	IRQ2_C	TO00_B	—	—	—	RXDA0_A	—
13	8	7	5	RES	P206	—	—	—	—	—	—	—
14	9	8	6	PCLBUZ0_A	P201	IRQ5_B	TI05_B/TO05_B	RTCCOUT_B	SSI00_B/ SCK11_B/ SCL11_B	—	—	—
15	10	9	7	—	P200	IRQ0_A/NMI	—	—	—	—	—	—
16	11	10	8	SWCLK	P300	—	TI04_B/TO04_B	—	—	—	—	—
17	12	11	9	SWDIO	P108	—	TI03_B/TO03_B	—	—	—	—	—
18	13	12	—	—	P109	IRQ4_B	TI02_A/TO02_A	—	TXD2_A/ SO20_A	SDAA0_C	TXDA0_C	—
19	14	13	—	—	P110	IRQ3_B	TI01_A/TO01_A	—	RXD2_A/ SI20_A/ SDA20_A	SCLA0_C	RXDA0_C	—
20	15	14	—	—	P112	IRQ2_B	TI03_A/TO03_A	—	SCK20_A/ SCL20_A/ SSI00_C	—	—	—
21	—	—	—	—	P103	IRQ5_A	TI05_A/TO05_A	—	SSI00_A	—	—	—
22	16	15	10	PCLBUZ0_B	P102	IRQ4_A	TI06_A/ TO06_A/ TO00_C	RTCCOUT_C	SCK00_A/ SCL00_A	—	—	—
23	17	16	11	—	P101	IRQ3_A	TI07_A/ TO07_A/TO00_C	—	TXD0_A/ SO00_A	SDAA0_D	TXDA0_D	AN021
24	18	17	12	—	P100	IRQ2_A	TI04_A/ TO04_A/ TI01_B/TO01_B	—	RXD0_A/ SI00_A/ SDA00_A	SCLA0_D	RXDA0_D	AN022
25	19	—	—	—	P015	IRQ1_A	—	—	—	—	—	AN007
26	20	—	—	—	P014	—	—	—	—	—	—	AN006
27	21	18	—	—	P013	—	—	—	—	—	—	AN005
28	22	19	13	—	P012	—	—	—	—	—	—	AN004
29	—	—	—	—	P009	—	—	—	—	—	—	AN003
30	—	—	—	—	P008	—	—	—	—	—	—	AN002
31	23	20	14	VREFL0	P011	—	—	—	—	—	—	AN001
32	24	1	15	VREFH0	P010	—	—	—	—	—	—	AN000

Note 1. When setting CMC.XTSEL = 1 for 24-, 20-, and 16-pin products

Note: Some signal names have A, B, C or D suffixes, but these suffixes can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for **SCK11**, **SCL11** and **SSI00**, bearing the same suffix can be selected. The simultaneous use of the same signal with different suffixes is prohibited.

[Before correction of 9.2.1 SBYCR : Standby Control Register] (Page 103)

Bit	Symbol	Function	R/W
15	SSBY	Software Standby Mode Select 0: Sleep mode 1: Software Standby mode.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

FLSTP bit (Flash Mode in Sleep Mode or in Snooze Mode)

When the FLSTP bit is set to 1, power consumption can be reduced because Flash stops during Sleep mode or Snooze mode. Instead, the Sleep mode release or Snooze mode release time will be extended. Also flash programming/erasure is not possible.

FWKUP bit (Setting for Starting the High-speed On-chip Oscillator at the times of release from Software Standby Mode and of Transitions to Snooze Mode)

When the FWKUP bit is set to 1, High-speed On-chip Oscillator enters high-speed startup mode, shortening standby release time and Snooze transition time. Instead, the frequency accuracy of HOCO changes during OSCSF.HOCOSF = 0 after ICLK starts operating, see [section 31, Electrical Characteristics](#). This bit can be set only when ICLK = HOCO (32MHz).

RTCLPC bit (SOSC Setting in Software Standby Mode or in Snooze Mode)

When the RTCLPC bit is set to 1, peripheral functions operating on SOSC other than RTC stop at Software Standby mode.

SSBY bit (Software Standby Mode Select)

The SSBY bit specifies the transition destination after a WFI instruction is executed.

When the SSBY bit is set to 1, the MCU enters Software Standby mode after execution of a WFI instruction. When the MCU returns to Normal mode from Software Standby mode by an interrupt, the SSBY bit remains 1. The SSBY bit can be cleared by writing 0 to it.

While the FENTRYR.FENTRY0 bit is 1 setting of the SSBY bit is ignored. Even if SSBY bit is 1, the MCU enters Sleep mode on execution of a WFI instruction.

(Omitted)

[After correction of 9.2.1 SBYCR : Standby Control Register]

Bit	Symbol	Function	R/W
15	SSBY	Software Standby Mode Select 0: Sleep mode 1: Software Standby mode.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

FLSTP bit (Flash Mode in Sleep Mode or in Snooze Mode)

When the FLSTP bit is set to 1, power consumption can be reduced because Flash stops during Sleep mode or Snooze mode. Instead, the Sleep mode release or Snooze mode release time will be extended. Also flash programming/erasure is not possible.

FWKUP bit (Setting for Starting the High-speed On-chip Oscillator at the times of release from Software Standby Mode and of Transitions to Snooze Mode)

When the FWKUP bit is set to 1, High-speed On-chip Oscillator enters high-speed startup mode, shortening standby release time and Snooze transition time. Instead, the frequency accuracy of HOCO changes during OSCSF.HOCOSF = 0 after ICLK starts operating, see [section 31, Electrical Characteristics](#). This bit can be set only when ICLK = HOCO (32MHz).

RTCLPC bit (SOSC Setting in Software Standby Mode or in Snooze Mode)

When the RTCLPC bit is set to 1, peripheral functions operating on SOSC other than RTC stop at Software Standby mode.

SSBY bit (Software Standby Mode Select)

The SSBY bit specifies the transition destination after a WFI instruction is executed.

When the SSBY bit is set to 1, the MCU enters Software Standby mode after execution of a WFI instruction. When the MCU returns to Normal mode from Software Standby mode by an interrupt, the SSBY bit remains 1. The SSBY bit can be cleared by writing 0 to it.

While the **FENTRYR.FENTRY0 bit or FENTRYR.FENTRYD** bit is 1 setting of the SSBY bit is ignored. Even if SSBY bit is 1, the MCU enters Sleep mode on execution of a WFI instruction.

(Omitted)

[Before correction of Table 16.6 Examples of register settings for port and alternate functions (2/6) (2 of 3)] (Page 196)

Table 16.6 Examples of register settings for port and alternate functions (2/6) (2 of 3)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P103	P103	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI06	Input	001b	0	0	0	0	x
	TO06	Output	001b	0	0	x	1	x
	SSI00	Input	010b	0	0	0	0	x
	IRQ4	Input	000b	1	0	0	0	x
P108	P108	Input	000b	—	—	0	0	x
		Output	000b	—	—	x	1	0/1
	SWDIO	I/O	001b	—	—	x	x	x
	TI03	Input	010b	—	—	0	0	x
	TO03	Output	010b	—	—	x	1	x
P109	P109	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI02	Input	001b	0	0	0	0	x
	TO02	Output	001b	0	0	x	1	x
	TXD2	Output	010b	0	0/1	0/1	1	x
	SO00	Output	010b	0	0/1	0/1	1	x
	SDAA0	I/O	011b	0	1	0	1	x
	TXDA0	Output	100b	0	0/1	0/1	1	x
IRQ3	Input	000b	1	0	0	0	x	
P110	P110	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI01	Input	001b	0	x	0	0	x
	TO01	Output	001b	0	0	x	1	x
	RXD2	Input	010b	0	x	0	0	x
	SI20	Input	010b	0	x	0	0	x
	SDA20	I/O	010b	0	1	0	1	x
	SCLA0	I/O	011b	0	1	0	1	x
	RXDA0	Input	100b	0	x	0	0	x
	IRQ3	Input	000b	1	0	0	0	x

[After correction of Table 16.6 Examples of register settings for port and alternate functions (2/6) (2 of 3)]

Table 16.6 Examples of register settings for port and alternate functions (2/6) (2 of 3)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P103	P103	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI05	Input	001b	0	0	0	0	x
	TO05	Output	001b	0	0	x	1	x
	SSI00	Input	010b	0	0	0	0	x
	IRQ5	Input	000b	1	0	0	0	x
P108	P108	Input	000b	—	—	0	0	x
		Output	000b	—	—	x	1	0/1
	SWDIO	I/O	001b	—	—	x	x	x
	TI03	Input	010b	—	—	0	0	x
	TO03	Output	010b	—	—	x	1	x
P109	P109	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI02	Input	001b	0	0	0	0	x
	TO02	Output	001b	0	0	x	1	x
	TXD2	Output	010b	0	0/1	0/1	1	x
	SO20	Output	010b	0	0/1	0/1	1	x
	SDAA0	I/O	011b	0	1	0	1	x
	TXDA0	Output	100b	0	0/1	0/1	1	x
IRQ4	Input	000b	1	0	0	0	x	
P110	P110	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI01	Input	001b	0	x	0	0	x
	TO01	Output	001b	0	0	x	1	x
	RXD2	Input	010b	0	x	0	0	x
	SI20	Input	010b	0	x	0	0	x
	SDA20	I/O	010b	0	1	0	1	x
	SCLA0	I/O	011b	0	1	0	1	x
	RXDA0	Input	100b	0	x	0	0	x
	IRQ3	Input	000b	1	0	0	0	x

[Before correction of Table 16.6 Examples of register settings for port and alternate functions (2/6) (3 of 3)] (Page 197)

Table 16.6 Examples of register settings for port and alternate functions (2/6) (3 of 3)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P112	P112	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI03	Input	001b	0	0	0	0	x
	TO03	Output	001b	0	0	x	1	x
	SCK20	Input	010b	0	x	0	0	x
		Output	010b	0	0/1	0/1	1	x
	SCL20	Output	010b	0	0/1	0/1	1	x
	SSI00	Input	011b	0	0	0	0	x
IRQ4	Input	000b	1	0	0	0	x	

(Omitted)

[After correction of Table 16.6 Examples of register settings for port and alternate functions (2/6) (3 of 3)]

Table 16.6 Examples of register settings for port and alternate functions (2/6) (3 of 3)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P112	P112	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI03	Input	001b	0	0	0	0	x
	TO03	Output	001b	0	0	x	1	x
	SCK20	Input	010b	0	x	0	0	x
		Output	010b	0	0/1	0/1	1	x
	SCL20	Output	010b	0	0/1	0/1	1	x
	SSI00	Input	011b	0	0	0	0	x
IRQ2	Input	000b	1	0	0	0	x	

(Omitted)

[Before correction of 16.6 Peripheral Select Settings for Each Product] (Page 200)

(Omitted)

16.6 Peripheral Select Settings for Each Product

This section describes the pin function select configuration by the PmnPFS_A register. Some signal names have _A, _B, _C, _D, _E, or _F suffixes, but these suffixes can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for SCL11 and SCK11, bearing the same suffix can be selected. The simultaneous use of the same signal with different suffixes is prohibited. Only the allowed values (functions) should be specified in the PSEL bits of PmnPFS_A. If a value that is not allowed for the register is specified, the correct operation is not guaranteed.

[After correction of 16.6 Peripheral Select Settings for Each Product]

(Omitted)

16.6 Peripheral Select Settings for Each Product

This section describes the pin function select configuration by the PmnPFS_A register. Some signal names have _A, _B, _C, _D, _E, or _F suffixes, but these suffixes can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for **SCK11, SCL11 and SCI00**, bearing the same suffix can be selected. The simultaneous use of the same signal with different suffixes is prohibited. Only the allowed values (functions) should be specified in the PSEL bits of PmnPFS_A. If a value that is not allowed for the register is specified, the correct operation is not guaranteed.

[Before correction of 22.2.7 IICWL0 : IICA Low-level Width Setting Register 0] (Page 475)

(Omitted)

22.2.7 IICWL0 : IICA Low-level Width Setting Register 0

Base address: IICA = 0x400A_3000

Offset address: 0x0102

Bit position: 7 6 5 4 3 2 1 0

Bit field:

Value after reset: 1 1 1 1 1 1 1 1

Bit	Symbol	Function	R/W
7:0	n/a	SCLA0 Pin Low-width Configuration Data of Unit n	R/W

This register is used to set the low-level width (t_{LOW}) of the SCLA0 pin signal that is output by I²C bus interface (IICA) and to control the SDAA0 pin signal.

Set the IICWL0 register while the I²C operation is disabled (IICCTL00.IICE).

For details about setting the IICWL0 register, see [section 22.3.2. Setting Transfer Clock Using IICWL0 and IICWH0 Registers](#). The data hold time is one-quarter of the time set by the IICWL0 register.

(Omitted)

[After correction of 22.2.7 IICWL0 : IICA Low-level Width Setting Register 0]

(Omitted)

22.2.7 IICWL0 : IICA Low-level Width Setting Register 0

Base address: IICA = 0x400A_3000

Offset address: 0x0102

Bit position: 7 6 5 4 3 2 1 0

Bit field:

Value after reset: 1 1 1 1 1 1 1 1

Bit	Symbol	Function	R/W
7:0	n/a	SCLA0 Pin Low-width Configuration Data of Unit n	R/W

This register is used to set the low-level width (t_{LOW}) of the SCLA0 pin signal that is output by I²C bus interface (IICA) and to control the SDAA0 pin signal.

Set the IICWL0 register while the I²C operation is disabled (IICCTL00.IICE).

For details about setting the IICWL0 register, see [section 22.3.2. Setting Transfer Clock Using IICWL0 and IICWH0 Registers](#). The data hold time is one-quarter of the time set by the IICWL0 register.

Note: The minimum serial clock cycle is $(IICWL0 + 1) + (IICWH0 + 1)$

(Omitted)

[Before correction of 22.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers] (Page 476)

(Omitted)

22.2.8 IICWH0 : IICA High-level Width Setting Register 0

Base address: IICA = 0x400A_3000

Offset address: 0x0103

Bit position: 7 6 5 4 3 2 1 0

Bit field:

Value after reset: 1 1 1 1 1 1 1 1

Bit	Symbol	Function	R/W
7:0	n/a	SCLA0 Pin High-width Configuration Data of Unit n	R/W

This register is used to set the high-level width of the SCLA0 pin signal that is output by I²C bus interface (IICA) and to control the SDAA0 pin signal.

Set the IICWH0 register while the I²C operation is disabled (IICCTL00.IICE).

For the procedures for setting the transfer clock in master mode and the IICWL0 and IICWH0 registers in slave mode, refer to [\(1\) Transfer clock setting in master mode](#) and [\(2\) Setting of IICWL0 and IICWH0 registers in slave mode](#), respectively.

(Omitted)

[After correction of 22.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers]

(Omitted)

22.2.8 IICWH0 : IICA High-level Width Setting Register 0

Base address: IICA = 0x400A_3000

Offset address: 0x0103

Bit position: 7 6 5 4 3 2 1 0

Bit field:

Value after reset: 1 1 1 1 1 1 1 1

Bit	Symbol	Function	R/W
7:0	n/a	SCLA0 Pin High-width Configuration Data of Unit n	R/W

This register is used to set the high-level width of the SCLA0 pin signal that is output by I²C bus interface (IICA) and to control the SDAA0 pin signal.

Set the IICWH0 register while the I²C operation is disabled (IICCTL00.IICE).

For the procedures for setting the transfer clock in master mode and the IICWL0 and IICWH0 registers in slave mode, refer to [\(1\) Transfer clock setting in master mode](#) and [\(2\) Setting of IICWL0 and IICWH0 registers in slave mode](#), respectively.

Note: The minimum serial clock cycle is (IICWL0 + 1) + (IICWH0 + 1)

(Omitted)

[Before correction of 22.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers] (Page 477)

(Omitted)

2.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers

(1) Transfer clock setting in master mode

$$\text{Transfer clock} = \frac{f_{MCK}}{IICWL + IICWH + f_{MCK}(t_R + t_F)}$$

At this time, the optimal setting values of the IICWL0 and IICWH0 registers are as follows. (The fractional parts of all setting values are rounded up.)

- When the fast mode

$$IICWL0 = \frac{0.52}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH0 = \left(\frac{0.48}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- When the normal mode

$$IICWL0 = \frac{0.47}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH0 = \left(\frac{0.53}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- When the fast mode plus

$$IICWL0 = \frac{0.50}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH0 = \left(\frac{0.50}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

(2) Setting of IICWL0 and IICWH0 registers in slave mode

The fractional parts of all setting values are rounded up.

- When the fast mode

$$IICWL0 = 1.3 \mu\text{s} \times f_{MCK}$$

$$IICWH0 = (1.2 \mu\text{s} - t_R - t_F) \times f_{MCK}$$

- When the normal mode

$$IICWL0 = 4.7 \mu\text{s} \times f_{MCK}$$

$$IICWH0 = (5.3 \mu\text{s} - t_R - t_F) \times f_{MCK}$$

- When the fast mode plus

$$IICWL0 = 0.50 \mu\text{s} \times f_{MCK}$$

$$IICWH0 = (0.50 \mu\text{s} - t_R - t_F) \times f_{MCK}$$

Note: Calculate the rise time (t_R) and fall time (t_F) of the SDAA0 and SCLA0 signals separately, because they differ depending on the pull-up resistance and wire load.

Note: IICWL0: IICA low-level width setting register 0

IICWH0: IICA high-level width setting register 0

t_F : SDAA0 and SCLA0 signal falling times

t_R : SDAA0 and SCLA0 signal rising times

f_{MCK} : IICA operation clock frequency

(Omitted)

[After correction of 22.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers]

(Omitted)

2.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers

(1) Transfer clock setting in master mode

$$\text{Transfer clock} = \frac{f_{MCK}}{IICWL + IICWH + f_{MCK}(t_R + t_F)}$$

At this time, the optimal setting values of the IICWL0 and IICWH0 registers are as follows. (The fractional parts of all setting values are rounded up.)

- When the fast mode

$$IICWL0 = \frac{0.52}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH0 = \left(\frac{0.48}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- When the normal mode

$$IICWL0 = \frac{0.47}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH0 = \left(\frac{0.53}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- When the fast mode plus

$$IICWL0 = \frac{0.50}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH0 = \left(\frac{0.50}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

(2) Setting of IICWL0 and IICWH0 registers in slave mode

The fractional parts of all setting values are rounded up.

- When the fast mode

$$IICWL0 = 1.3 \mu\text{s} \times f_{MCK}$$

$$IICWH0 = (1.2 \mu\text{s} - t_R - t_F) \times f_{MCK}$$

- When the normal mode

$$IICWL0 = 4.7 \mu\text{s} \times f_{MCK}$$

$$IICWH0 = (5.3 \mu\text{s} - t_R - t_F) \times f_{MCK}$$

- When the fast mode plus

$$IICWL0 = 0.50 \mu\text{s} \times f_{MCK}$$

$$IICWH0 = (0.50 \mu\text{s} - t_R - t_F) \times f_{MCK}$$

Note: Calculate the rise time (t_R) and fall time (t_F) of the SDAA0 and SCLA0 signals separately, because they differ depending on the pull-up resistance and wire load.

Note: IICWL0: IICA low-level width setting register 0

IICWH0: IICA high-level width setting register 0

t_F : SDAA0 and SCLA0 signal falling times

t_R : SDAA0 and SCLA0 signal rising times

f_{MCK} : IICA operation clock frequency

Note: The minimum serial clock cycle is $(IICWL0 + 1) + (IICWH0 + 1)$. Determine the values to be set in the IICWL0 and IICWH0 registers by considering the rise time (t_R) and fall time (t_F) of the SDAA0 and SCLA0 signals.

(Omitted)

[Before correction of 22.3.10 Timing of Generation of the Interrupt Request Signal (IICAn_TXRXI) and Control of Clock Stretching] (Page 482, 483)

(Omitted)

22.3.10 Timing of Generation of the Interrupt Request Signal (IICA0_TXRXI) and Control of Clock Stretching

The setting of WTIM bit of IICA control register 00 (IICCTL00) determines the timing by which IICA0_TXRXI is generated and controls clock stretching, as shown in Table 22.2.

The numbers in the table indicate the pulses of the serial clock signal. Interrupt requests and control of clock stretching are both synchronized with the falling edge of these clock pulses.

Table 22.2 IICA0_TXRXI generation timing and control of clock stretching

WTIM	During slave device operation			During master device operation		
	Address	Data reception	Data transmission	Address	Data reception	Data transmission
0	9 ^{*1} *2	8 ^{*2}	8 ^{*2}	9	8	8
1	9 ^{*1} *2	9 ^{*2}	9 ^{*2}	9	9	9

1. During address transmission/reception

- Slave device operation: The timing of the interrupt and clock stretching depends on the conditions described in ^{*1} and ^{*2} above, regardless of the setting of the IICCTL00.WTIM bit.
- Master device operation: The interrupt and clock stretching occur at the falling edge of the ninth clock cycle, regardless of the setting of the IICCTL00.WTIM bit.

2. During data reception

- All operation: The timing of the interrupt and clock stretching depends on the setting of the IICCTL00.WTIM bit.

3. During data transmission

- All operation: The timing of the interrupt and clock stretching depends on the setting of the IICCTL00.WTIM bit.

4. Release from clock stretching

The four types of processing for release from clock stretching are as follows.

- Writing data to the IICA shift register 0 (IICA0)
- Setting WREL bit of IICA control register 00 (IICCTL00) (release from the clock stretch state)
- Setting STT bit of the IICCTL00 register (generating start condition)^{*3}
- Setting SPT bit of the IICCTL00 register (generating stop condition)^{*3}

When 8th cycle clock stretching has been selected (IICCTL00.WTIM = 0), the presence or absence of ACK generation must be determined before release from the clock stretch state.

5. Detection of stop condition

IICA0_TXRXI is generated when a stop condition is detected (only when IICCTL00.SPIE = 1).

Note 1. The IICA0_TXRXI signal of the slave device and clock stretching occur at the falling edge of the 9th clock cycle only when there is a match with the address set to the slave address register 0 (SVA0).

At this point, ACK is generated regardless of the value set to the IICCTL00.ACKE bit. For a slave device that has received an extension code, or has received an address while the all address match function is enabled, IICA0_TXRXI occurs at the falling edge of the eighth clock.

However, if the address does not match after restart, IICA0_TXRXI is generated at the falling edge of the 9th clock cycle, but clock stretching does not occur.

Note 2. When the Slave Address Register n (SVAn) does not match the received address, the address match function is disabled, and an extended code has not been received, neither IICA0_TXRXI nor clock stretching will occur.

Note 3. Master mode only

(Omitted)

[After correction of 22.3.10 Timing of Generation of the Interrupt Request Signal (IICAn_TXRXI) and Control of Clock Stretching]

(Omitted)

22.3.10 Timing of Generation of the Interrupt Request Signal (IICA0_TXRXI) and Control of Clock Stretching

The setting of WTIM bit of IICA control register 00 (IICCTL00) determines the timing by which IICA0_TXRXI is generated and controls clock stretching, as shown in Table 22.2.

The numbers in the table indicate the pulses of the serial clock signal. Interrupt requests and control of clock stretching are both synchronized with the falling edge of these clock pulses.

Table 22.2 IICA0_TXRXI generation timing and control of clock stretching

WTIM	During slave device operation			During master device operation		
	Address	Data reception	Data transmission	Address	Data reception	Data transmission
0	9 ^{*1} *2	8 ^{*2}	8 ^{*2}	9	8	8
1	9 ^{*1} *2 *3	9 ^{*2}	9 ^{*2}	9	9	9

1. During address transmission/reception

- Slave device operation: The timing of the interrupt and clock stretching depends on the conditions described in ^{*1} and ^{*2}, ^{*3} below, regardless of the setting of the IICCTL00.WTIM bit.
- Master device operation: The interrupt and clock stretching occur at the falling edge of the ninth clock cycle, regardless of the setting of the IICCTL00.WTIM bit.

2. During data reception

- All operation: The timing of the interrupt and clock stretching depends on the setting of the IICCTL00.WTIM bit.

3. During data transmission

- All operation: The timing of the interrupt and clock stretching depends on the setting of the IICCTL00.WTIM bit.

4. Release from clock stretching

The four types of processing for release from clock stretching are as follows.

- Writing data to the IICA shift register 0 (IICA0)
- Setting WREL bit of IICA control register 00 (IICCTL00) (release from the clock stretch state)
- Setting STT bit of the IICCTL00 register (generating start condition)^{*4}
- Setting SPT bit of the IICCTL00 register (generating stop condition)^{*4}

When 8th cycle clock stretching has been selected (IICCTL00.WTIM = 0), the presence or absence of ACK generation must be determined before release from the clock stretch state.

5. Detection of stop condition

IICA0_TXRXI is generated when a stop condition is detected (only when IICCTL00.SPIE = 1).

Note 1. The IICA0_TXRXI signal of the slave device and clock stretching occur at the falling edge of the 9th clock cycle only when there is a match with the address set to the slave address register 0 (SVA0).

At this point, ACK is generated regardless of the value set to the IICCTL00.ACKE bit. For a slave device that has received an extension code, or has received an address while the all address match function is enabled, IICA0_TXRXI occurs at the falling edge of the eighth clock.

However, if the address does not match after restart, IICA0_TXRXI is generated at the falling edge of the 9th clock cycle, but clock stretching does not occur.

Note 2. When the Slave Address Register n (SVAn) does not match the received address, the address match function is disabled, and an extended code has not been received, neither IICA0_TXRXI nor clock stretching will occur.

Note 3. When the setting of the WTIMn bit is 1, for a slave device that has received an extension code, or has received an address while the all address match function is enabled, INTIICAn and clock stretching occur at the falling edge of the eighth clock as well as on that of the ninth clock.

Note 4. Master mode only

(Omitted)

[Before correction of 23.3.4 Baud Rate Generator] (Page 548, 549)

(Omitted)

After the start bit is detected, the latch timing of receive data is determined by the counter specified with the baud rate generator control register (BRGCA0). If the whole frame including the stop bit has been received before this latching, reception can proceed correctly.

Assuming that 11 bits of data are received, the theoretical values can be calculated as follows.

- The relation between 1-bit data length and baud rate
 $FL = (\text{Brate}) - 1$
 Brate: Baud rate of UART
 k: Set value of BRGCA0 FL: 1-bit data length
 Margin of latch timing: 1 clock
- Minimum permissible data frame length (FLmin)
 $FL_{\min} = 11 \times FL - \frac{k-1}{2k} \times FL = \frac{21k+1}{2k} FL$
- Maximum permissible baud rate for reception on the transmitting side (BRmax)
 $BR_{\max} = (FL_{\min}/11)^{-1} = \frac{22k}{21k+1} \text{Brate}$
- Maximum permissible data frame length (FLmax)
 $FL_{\max} = \frac{21k+1}{20k} FL \times 11$
- Minimum permissible baud rate for reception on the transmitting side (BRmin)
 $BR_{\min} = (FL_{\max}/11)^{-1} = \frac{20k}{21k-1} \text{Brate}$

Table 23.13 shows the permissible baud rate error between UART and the transmitting side can be calculated from the above minimum and maximum baud rate expressions.

Table 23.13 Maximum and minimum permissible baud rate error

Division ratio (k)	Maximum permissible baud rate error	Minimum permissible baud rate error
2	+2.32%	-2.43%
4	+3.52%	-3.61%
8	+4.14%	-4.19%
20	+4.51%	-4.53%
50	+4.66%	-4.67%
100	+4.71%	-4.71%
255	+4.74%	-4.74%

Note: The permissible error of reception depends on the number of bits in one frame, input clock frequency, and division ratio (k).

The higher the input clock frequency and the division ratio (k), the higher the permissible error.

Note: k: Set value of BRGCA0

(Omitted)

[After correction of 23.3.4 Baud Rate Generator]

(Omitted)

After the start bit is detected, the latch timing of receive data is determined by the counter specified with the baud rate generator control register (BRGCA0). If the whole frame including the stop bit has been received before this latching, reception can proceed correctly.

Assuming that 11 bits of data are received, the theoretical values can be calculated as follows.

- The relation between 1-bit data length and baud rate
 $FL = (Brate) - 1$
 Brate: Baud rate of UART
 k: Set value of BRGCA0 FL: 1-bit data length
 Margin of latch timing: 2 clock
- Minimum permissible data frame length (FLmin)
 $k = 3 \text{ to } 255: FL_{min} = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$
- Maximum permissible baud rate for reception on the transmitting side (BRmax)
 $k = 2: BR_{max} = Brate + \frac{1}{22k} Brate$
 $k = 3 \text{ to } 255: BR_{max} = (FL_{min}/11)^{-1} = \frac{22k}{21k+2} Brate$
- Maximum permissible data frame length (FLmax)
 $k = 3 \text{ to } 255: FL_{max} = \frac{21k+2}{20k} FL \times 11$
- Minimum permissible baud rate for reception on the transmitting side (BRmin)
 $k = 2: BR_{min} = Brate + \frac{1}{22k} Brate$
 $k = 3 \text{ to } 255: BR_{min} = (FL_{max}/11)^{-1} = \frac{20k}{21k-2} Brate$

Table 23.13 shows the permissible baud rate error between UART and the transmitting side can be calculated from the above minimum and maximum baud rate expressions.

Table 23.13 Maximum and minimum permissible baud rate error

Division ratio (k)	Maximum permissible baud rate error	Minimum permissible baud rate error
2	+2.27%	-2.27%
4	+2.33%	-2.44%
8	+3.53%	-3.61%
20	+4.27%	-4.31%
50	+4.56%	-4.58%
100	+4.66%	-4.67%
255	+4.72%	-4.73%

Note: The permissible error of reception depends on the number of bits in one frame, input clock frequency, and division ratio (k).

The higher the input clock frequency and the division ratio (k), the higher the permissible error.

Note: k: Set value of BRGCA0

(Omitted)

[Before correction of 28.3.25 PNRn : Part Numbering Register n (n = 0 to 3)] (Page 636)

(Omitted)

28.3.25 PNRn : Part Numbering Register n (n = 0 to 3)

Address: 0x0101_1080 + n × 4

Bit position: 31 0



Value after reset: Unique value for each chip

Bit	Symbol	Function	R/W
31:0	PNR[31:0]	Part Number	R

The PNRn is a read-only register that stores a 16-byte part numbering. The PNRn register should be read in 32-bit units. Each byte corresponds to the ASCII code representation of the product part number as detailed in product list.

In case of the part number is 'R7FA0E1073CNK', 16-byte part numbering is stored as follows.

Address 0x0101_1080: 'K', 0x4B in ASCII code

Address 0x0101_1081: 'N', 0x4E in ASCII code

Address 0x0101_1082: 'C', 0x43 in ASCII code

Address 0x0101_1083: '3', 0x33 in ASCII code

Address 0x0101_1084: '7', 0x37 in ASCII code

Address 0x0101_1085: '0', 0x30 in ASCII code

Address 0x0101_1086: '1', 0x31 in ASCII code

Address 0x0101_1087: 'E', 0x45 in ASCII code

Address 0x0101_1088: '0', 0x30 in ASCII code

Address 0x0101_1089: 'A', 0x41 in ASCII code

Address 0x0101_1090: 'F', 0x46 in ASCII code

Address 0x0101_1091: '7', 0x37 in ASCII code

Address 0x0101_1092: 'R', 0x52 in ASCII code

Address 0x0101_1093: '(space)', 0x20 in ASCII code

Address 0x0101_1094: '(space)', 0x20 in ASCII code

Address 0x0101_1095: '(space)', 0x20 in ASCII code

(Omitted)

[After correction of 28.3.25 PNRn : Part Numbering Register n (n = 0 to 3)]

(Omitted)

28.3.25 PNRn : Part Numbering Register n (n = 0 to 3)

Address: 0x0101_1080 + n × 4

Bit position: 31

0

Bit field:

PNR[31:0]

Value after reset:

Unique value for each chip

Bit	Symbol	Function	R/W
31:0	PNR[31:0]	Part Number	R

The PNRn is a read-only register that stores a 16-byte part numbering. The PNRn register should be read in 32-bit units. Each byte corresponds to the ASCII code representation of the product part number as detailed in product list.

In case of the part number is 'R7FA0E1073CNK', 16-byte part numbering is stored as follows.

Address 0x0101_1080: 'K', 0x4B in ASCII code

Address 0x0101_1081: 'N', 0x4E in ASCII code

Address 0x0101_1082: 'C', 0x43 in ASCII code

Address 0x0101_1083: '3', 0x33 in ASCII code

Address 0x0101_1084: '7', 0x37 in ASCII code

Address 0x0101_1085: '0', 0x30 in ASCII code

Address 0x0101_1086: '1', 0x31 in ASCII code

Address 0x0101_1087: 'E', 0x45 in ASCII code

Address 0x0101_1088: '0', 0x30 in ASCII code

Address 0x0101_1089: 'A', 0x41 in ASCII code

Address 0x0101_108A: 'F', 0x46 in ASCII code

Address 0x0101_108B: '7', 0x37 in ASCII code

Address 0x0101_108C: 'R', 0x52 in ASCII code

Address 0x0101_108D: '(space)', 0x20 in ASCII code

Address 0x0101_108E: '(space)', 0x20 in ASCII code

Address 0x0101_108F: '(space)', 0x20 in ASCII code

(Omitted)

[Before correction of Table 31.12 Operating and standby current (1) (2 of 2)] (Page 672)

Table 31.12 Operating and standby current (1) (2 of 2)

Conditions: VCC = 1.6 to 5.5 V

Parameter					Symbol	Typ ⁵	Max	Unit	Test Conditions	
Supply current ^{*1}	Subosc-speed mode ^{*4}	Normal mode	Peripheral clocks disabled	ICLK = 32.768 kHz	Ta = -40°C	3.3	—	μA	—	
					Ta = 25°C	3.7	—			
					Ta = 50°C	3.9	—			
					Ta = 70°C	4.3	—			
					Ta = 85°C	4.8	—			
					Ta = 105°C	6.2	—			
			Peripheral clocks enabled ^{*7}	Ta = -40°C	—	7.2				
				Ta = 25°C	—	7.9				
				Ta = 50°C	—	9.6				
				Ta = 70°C	—	13.0				
	Sleep mode		Peripheral clocks disabled	ICLK = 32.768 kHz	Ta = -40°C	1.0	—		—	
					Ta = 25°C	1.3	—			
					Ta = 50°C	1.5	—			
					Ta = 70°C	1.8	—			
					Ta = 85°C	2.2	—			
					Ta = 105°C	3.2	—			
					Peripheral clocks enabled ^{*7}	Ta = -40°C	—			4.8
						Ta = 25°C	—			5.4
						Ta = 50°C	—			7.0
						Ta = 70°C	—			10.5
		Peripheral clocks enabled ^{*7}	ICLK = 32.768 kHz	Ta = 85°C	—	16.1				
				Ta = 105°C	—	33.3				

- Note 1. Supply current is the total current flowing into VCC. Supply current values apply when internal pull-up MOSs are in the off state and these values do not include output charge/discharge current from any of the pins.
- Note 2. The clock source is high-speed on-chip oscillator (HOCO).
- Note 3. The clock source is middle-speed on-chip oscillator (MOCO).
- Note 4. The clock source is the Sub-clock oscillator (SOSC) and CMC.SODRV[1:0] are 10b (Low power mode 2).
- Note 5. VCC = 3.3 V.
- Note 6. Includes operating current for PCLBUZ, TAU, SAU, and IICA functions only. For other peripheral operating currents, please add the current in Peripheral Functions Supply current in [Table 31.14](#).
- Note 7. Includes operating current for PCLBUZ, TAU and SAU functions only. For other peripheral operating currents, please add the current in [Table 31.14](#).

[After correction of Table 31.12 Operating and standby current (1) (2 of 2)]

Table 31.12 Operating and standby current (1) (2 of 2)

Conditions: VCC = 1.6 to 5.5 V

Parameter					Symbol	Typ ⁵	Max	Unit	Test Conditions		
Supply current ^{*1}	Subosc-speed mode ^{*4}	Normal mode	Peripheral clocks disabled	ICLK = 32.768 kHz	Ta = -40°C	3.3	—	μA	—		
					Ta = 25°C	3.7	—				
					Ta = 50°C	3.9	—				
					Ta = 70°C	4.3	—				
					Ta = 85°C	4.8	—				
					Ta = 105°C	6.2	—				
			Peripheral clocks enabled ^{*7}	ICLK = 32.768 kHz	Ta = -40°C	—	7.2				
				Ta = 25°C	—	7.9					
				Ta = 50°C	—	9.6					
				Ta = 70°C	—	13.0					
	Sleep mode		Peripheral clocks disabled	ICLK = 32.768 kHz	Ta = -40°C	1.0	—		—		
					Ta = 25°C	1.3	—				
					Ta = 50°C	1.5	—				
					Ta = 70°C	1.8	—				
					Ta = 85°C	2.2	—				
					Ta = 105°C	3.2	—				
					Peripheral clocks enabled ^{*7}	ICLK = 32.768 kHz	Ta = -40°C			—	4.8
						Ta = 25°C	—			5.4	
						Ta = 50°C	—			7.0	
						Ta = 70°C	—			10.5	
				Ta = 85°C	—	16.1					
				Ta = 105°C	—	33.3					

- Note 1. Supply current is the total current flowing into VCC. Supply current values apply when internal pull-up MOSs are in the off state and these values do not include output charge/discharge current from any of the pins.
- Note 2. The clock source is high-speed on-chip oscillator (HOCO).
- Note 3. The clock source is middle-speed on-chip oscillator (MOCO).
- Note 4. The clock source is the Sub-clock oscillator (SOSC) and CMC.SODRV[1:0] are 10b (Low power mode 2).
- Note 5. VCC = 3.3 V.
- Note 6. Includes operating current for PCLBUZ, TAU, SAU, and IICA functions only. For other peripheral operating currents, please add the current in Peripheral Functions Supply current in [Table 31.14](#).
- Note 7. Includes operating current for PCLBUZ, TAU and SAU functions only. For other peripheral operating currents, please add the current in [Table 31.14](#).
- Note 8. The operating conditions under which the sub-clock oscillator can be used are specified separately. For details, see section 31.2.2. [Sub-clock Oscillator Characteristics](#).