Date: Jul. 4, 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-RA*-A0114B/E	Rev.	2.00
Title	Updates regarding RA8D1 Group user's manual		Information Category	Technical Notification		
Applicable Product	RA8D1 Group	Lot No. All	Reference Document	RA8D1 Group User's Manual : Hardware Rev.1.10		

Some descriptions in the RA8D1 Group User's Manual: Hardware, Rev.1.10 are corrected as follows:

Contents

1.Overview

Table 1.8 Communication interfaces

[Page 75]

Before correction

Octal Serial Peripheral Interface (OSPI)	The Octal Serial Peripheral Interface (OSPI) is a memory controller that supports Expanded Serial Peripheral Interface (xSPI) (JEDEC Standard JESD251, JESD251-1 and JESD252) . The OSPI supports 1-bit, 2-bit, 4-bit and 8-bit protocols. JESD251 specifies two interface profiles where profile 1.0 is Octal SPI and profile 2.0 is HyperBus™. See section 37, Octal Serial Peripheral Interface (OSPI).

After correction

Octal Serial Peripheral Interface (OSPI)	The Octal Serial Peripheral Interface (OSPI) is a memory controller that supports Expanded Serial Peripheral Interface (xSPI) (JEDEC Standard JESD251, JESD251-1 and JESD252) . The OSPI supports 1-bit, 2-bit, 4-bit and 8-bit protocols. JESD251 specifies two interface profiles where profile 1.0 is Octal SPI and profile 2.0 is HyperBus™ (HyperRAM™ and HyperFlash™). OSPI supports QSPI protocol. See section 37, Octal Serial Peripheral Interface (OSPI).

5. Resets

Table 5.3 Module-related registers initialized by each reset source (1 of 4)

[Page 141]

Before correction

Registers to be initialized		Reset source						
		RES# pin reset	**Power-on reset**	**Voltage monitoring 0 reset**	**Independent watchdog timer reset**	**Watchdog timer reset**	**CPU Lockup reset**	**Voltage monitoring 1 reset**
Voltage Monitor Function 1 registers	PVD1CR0, PVD1CMPCR, PVD1FCR	✓	✓	✓	✓	✓	—	—
	PVD1CR1. PVD1SR	✓	✓	✓	✓	✓	—	—
Voltage Monitor Function 2 registers	PVD2CR0, PVD2CMPCR, PVD2FCR	✓	✓	✓	✓	✓	—	—
	PVD2CR1. PVD2SR	✓	✓	✓	✓	✓	—	—
SOSC registers	SOSCCR, SOMCR	—	—	—	—	—	—	—
LOCO registers	LOCOUTCR	—	✓	✓	—	—	—	—
MOSC registers	MOMCR	✓	✓	✓	✓	✓	✓	✓
Pin states (except XCIN / XCOUT)		✓	✓	✓	✓	✓	✓	✓
Pin states (XCIN / XCOUT)		—	—	—	—	—	—	—
IO capture and tamper detection such as VBAT (sampling timing for RTC) （RTCIC0-2）	VBTICTLR,VBTICTLR2,VBTADSR、VBTADCR1,VBTADCR2	—	—	—	—	—	—	—

VBATT Battery power supply switch control register 1	VBTBPCR1	✓	✓	✓	✓	✓	✓	✓
VBATT Battery Power Supply Switch Control register 2	VBTBPCR2	—	—	—	—	—	—	—
VBATT Backup Enable register	VBTBER	—	✓	—	—	—	—	—
Battery Backup register	VBTBKR[n]	—	—	—	—	—	—	—

After correction

Registers to be initialized		Reset source						
		RES# pin reset	Power-on reset	Voltage monitoring 0 reset	Independent watchdog timer reset	Watchdog timer reset	CPU Lockup reset	Voltage monitoring 1 reset
Voltage Monitor Function 1 registers	PVD1CR0, PVD1CMPCR, PVD1FCR	✓	✓	✓	✓	✓	—	—
	PVD1CR1. PVD1SR	✓	✓	✓	✓	✓	—	—
Voltage Monitor Function 2 registers	PVD2CR0, PVD2CMPCR, PVD2FCR	✓	✓	✓	✓	✓	—	—
	PVD2CR1. PVD2SR	✓	✓	✓	✓	✓	—	—
SOSC registers	SOSCCR, SOMCR	—	—	—	—	—	—	—
LOCO registers	LOCOUTCR	—	✓	✓	—	—	—	—
MOSC registers	MOMCR	✓	✓	✓	✓	✓	✓	✓
HOCO control register 2	HOCOCR2	—	✓	✓	—	—	—	—
Pin states (except XCIN / XCOUT)		✓	✓	✓	✓	✓	✓	✓
Pin states (XCIN / XCOUT)		—	—	—	—	—	—	—
IO capture and tamper detection such as VBAT (sampling timing for RTC) （RTCIC0-2）	VBTICTLR,VBTICTLR2,VBTADSR、VBTADCR1,VBTADCR2	—	—	—	—	—	—	—
VBATT Battery power supply switch control register 1	VBTBPCR1	✓	✓	✓	✓	✓	✓	✓
VBATT Battery Power Supply Switch Control register 2	VBTBPCR2	—	—	—	—	—	—	—
VBATT Backup Enable register	VBTBER	—	✓	—	—	—	—	—
Battery Backup register	VBTBKR[n]	—	—	—	—	—	—	—

Table 5.3 Module-related registers initialized by each reset source (3 of 4)

[Page 143]

Before correction

Registers to be initialized		Reset source							
		Voltage monitoring 2 reset	Software reset	Bus error reset	Common memory error reset	Deep software standby reset			VBATT-selected voltage power-on reset
						Deep Software Standby mode 1 reset	Deep Software Standby mode 2 reset	Deep Software Standby mode 3 reset	
Voltage Monitor Function 1 registers	PVD1CR0, PVD1CMPCR, PVD1FCR	—	—	—	—	—	—	—	—
	PVD1CR1. PVD1SR	—	—	—	—	✓	✓	✓	—
Voltage Monitor Function 2 registers	PVD2CR0, PVD2CMPCR, PVD2FCR	—	—	—	—	—	—	—	—
	PVD2CR1. PVD2SR	—	—	—	—	✓	✓	✓	—
SOSC registers	SOSCCR, SOMCR	—	—	—	—	—	—	—	✓
LOCO registers	LOCOUTCR	—	—	—	—	—	✓	✓	—
MOSC registers	MOMCR	✓	✓	✓	✓	—	—	—	—
Pin states (except XCIN / XCOUT)		✓	✓	✓	✓	✓ *2	✓ *2	✓ *2	—
Pin states (XCIN / XCOUT)		—	—	—	—	—	—	—	●
IO capture and tamper detection such as VBAT (sampling timing for RTC) （RTCIC0-2）	VBTICTLR,VBTICTLR2,VBTADSR、VBTADCR1,VBTADCR2	—	—	—	—	—	—	—	●
VBATT Battery Power Supply Switch Control register 1	VBTBPCR1	✓	✓	✓	✓	—	—	—	—
VBATT Battery Power Supply Switch Control register 2	VBTBPCR2	—	—	—	—	—	—	—	✓
VBATT Backup Enable register	VBTBER	—	—	—	—	—	—	—	—
Battery Backup register	VBTBKR[n]	—	—	—	—	—	—	—	✓
VBATT Input Monitor Register	VBTIMONR	—	—	—	—	—	—	—	—
Independent Watchdog Timer registers	IWDTRR、IWDTCR、IWDTSR、IWDTRCR、IWDTCSTPR	✓	✓	✓	✓	—	✓	✓	—
Realtime Clock*1 register		—	—	—	—	—	—	—	—

After correction

Registers to be initialized		Reset source							
		Voltage monitoring 2 reset	Software reset	Bus error reset	Common memory error reset	Deep software standby reset			VBATT-selected voltage power-on reset
						Deep Software Standby mode 1 reset	Deep Software Standby mode 2 reset	Deep Software Standby mode 3 reset	
Voltage Monitor Function 1 registers	PVD1CR0, PVD1CMPCR, PVD1FCR	—	—	—	—	—	—	—	—
	PVD1CR1. PVD1SR	—	—	—	—	✓	✓	✓	—
Voltage Monitor Function 2 registers	PVD2CR0, PVD2CMPCR, PVD2FCR	—	—	—	—	—	—	—	—
	PVD2CR1. PVD2SR	—	—	—	—	✓	✓	✓	—
SOSC registers	SOSCCR, SOMCR	—	—	—	—	—	—	—	✓
LOCO registers	LOCOUTCR	—	—	—	—	—	✓	✓	—
MOSC registers	MOMCR	✓	✓	✓	✓	—	—	—	—
HOCO control register 2	HOCOCR2	—	—	—	—	✓ *3	✓ *3	✓ *3	—
Pin states (except XCIN / XCOUT)		✓	✓	✓	✓	✓ *2	✓ *2	✓ *2	—
Pin states (XCIN / XCOUT)		—	—	—	—	—	—	—	●
IO capture and tamper detection such as VBAT (sampling timing for RTC) （RTCIC0-2）	VBTICTLR,VBTICTLR2,VBTADSR、VBTADCR1,VBTADCR2	—	—	—	—	—	—	—	●
VBATT Battery Power Supply Switch Control register 1	VBTBPCR1	✓	✓	✓	✓	—	—	—	—
VBATT Battery Power Supply Switch Control register 2	VBTBPCR2	—	—	—	—	—	—	—	✓
VBATT Backup Enable register	VBTBER	—	—	—	—	—	—	—	—
Battery Backup register	VBTBKR[n]	—	—	—	—	—	—	—	✓
VBATT Input Monitor Register	VBTIMONR	—	—	—	—	—	—	—	—
Independent Watchdog Timer registers	IWDTRR、IWDTCR、IWDTSR、IWDTRCR、IWDTCSTPR	✓	✓	✓	✓	—	✓	✓	—
Realtime Clock*1 register		—	—	—	—	—	—	—	—

RENESAS

6. Option-Setting Memory

6.2.5 OFS1, OFS1_SEC : Option Function Select Register 1 for Non-secure and Secure

[Page 166]

Before correction

HOCOEN bit (HOCO Oscillation Enable)

The HOCOEN bit selects whether the HOCO oscillation is enabled or disabled after a reset. Setting this bit to 0 allows the HOCO oscillation to start before the CPU starts operation, which reduces the wait time for oscillation stabilization.

Note: When the HOCOEN bit is set to 0, the system clock source is not switched to HOCO. The system clock source is only switched to HOCO by setting the Clock Source Select bits (SCKSCR.CKSEL[2:0]). To use the HOCO clock, Set the OFS1(_SEC).HOCOFRQ0 bit to an optimum value.

(omission)

INITECCEN bit (Initial ECC Enable)

The INITECCEN bit selects whether ECC function of TCM and CACHE is enabled or disabled.

After correction

HOCOEN bit (HOCO Oscillation Enable)

The HOCOEN bit selects whether the HOCO oscillation is enabled or disabled after a reset. Setting this bit to 0 allows the HOCO oscillation to start before the CPU starts operation, which reduces the wait time for oscillation stabilization.

Note: When the HOCOEN bit is set to 0, the system clock source is not switched to HOCO. The system clock source is only switched to HOCO by setting the Clock Source Select bits (SCKSCR.CKSEL[2:0]). To use the HOCO clock, Set the OFS1(_SEC). HOCOFRQ0[2:0] bits *1 to an optimum value.

Note 1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits when OFS1(_SEC).HOCOEN=1.

(omission)

INITECCEN bit (Initial ECC Enable)

The INITECCEN bit selects whether ECC function of TCM and CACHE is enabled or disabled.

When the INITECCEN bit is changed from 1 to 0, be sure to perform a power-on reset after changing it.

8. Clock Generation Circuit

Table 8.1 Clock generation circuit specifications for the clock sources

[Page 195]

Before correction

PLL1 circuit PLL2 circuit	Input clock source	MOSC/HOCO
	Input pulse frequency division ratio	Selectable from 1/2/3/4
	Input clock frequency	8 MHz to 48 MHz
	Input clock frequency (After input frequency division)	6 MHz to 12 MHz
	Frequency multiplication ratio	Selectable from 26 to 180 (after the decimal point : 0/0.33/0.50/0.66)
	VCO frequency	640 MHz to 1440 MHz

	Number of output clocks	Output 3 different clocks
	PLL Output clock P	40 MHz to 480 MHz (output division ratio : 2/4/6/8/16)
	PLL Output clock Q	71 MHz to 480 MHz (output division ratio : 2/3/4/5/6/8/9)
	PLL Output clock R	71 MHz to 480 MHz (output division ratio : 2/3/4/5/6/8/9)

After correction

PLL1 circuit PLL2 circuit	Input clock source	MOSC/HOCO
	Input pulse frequency division ratio	Selectable from 1/2/3/4
	Input clock frequency	8 MHz to 48 MHz
	Input clock frequency (After input frequency division)	6 MHz to 12 MHz
	Frequency multiplication ratio	Selectable from 53 to 180 (after the decimal point : 0/0.33/0.50/0.66)
	VCO frequency	640 MHz to 1440 MHz
	Number of output clocks	Output 3 different clocks
	PLL Output clock P	40 MHz to 480 MHz (output division ratio : 2/4/6/8/16)
	PLL Output clock Q	71 MHz to 480 MHz (output division ratio : 2/3/4/5/6/8/9)
	PLL Output clock R	71 MHz to 480 MHz (output division ratio : 2/3/4/5/6/8/9)

8.2.1 CGFSAR : Clock Generation Function Security Attribute Register

[Page 200, 202]

Before correction

Bit	Symbol	Function	R/W
0	NONSEC00	Non Secure Attribute bit 00 Target register: SCKDIVCR, SCKDIVCR2, SCKSCR Target factor: system clock control 0: Secure 1: Non Secure	R/W
1	—	This bit is read as 0. The write value should be 0.	R/W
2	NONSEC02	Non Secure Attribute bit 02 Target register: HOCOCR, FLLCR1, FLLCR2, HOCOUTCR, HOCOSCR Target factor: HOCO 0: Secure 1: Non Secure	R/W

(omission)

NONSEC02 bit (Non Secure Attribute bit 02)

This bit controls the security attribute of HOCOCR, FLLCR1, FLLCR2, HOCOUTCR, HOCOSCR.

After correction

Bit	Symbol	Function	R/W
0	NONSEC00	Non Secure Attribute bit 00 Target register: SCKDIVCR, SCKDIVCR2, SCKSCR Target factor: system clock control 0: Secure 1: Non Secure	R/W
1	—	This bit is read as 0. The write value should be 0.	R/W
2	NONSEC02	Non Secure Attribute bit 02 Target register: HOCOCR, HOCOCR2, FLLCR1, FLLCR2, HOCOUTCR, HOCOSCR Target factor: HOCO 0: Secure 1: Non Secure	R/W

(omission)

NONSEC02 bit (Non Secure Attribute bit 02)

This bit controls the security attribute of HOCOCR, HOCOCR2, FLLCR1, FLLCR2, HOCOUTCR, HOCOSCR.

8.2.6 PLLCCR : PLL Clock Control Register

[Page 208]

Before correction

Bit	Symbol	Function	R/W
1:0	PLIDIV[1:0]*1	PLL1 Input Frequency Division Ratio Select 0 0: 1/1 0 1: 1/2 1 0: 1/3 1 1: 1/4	R/W
3:2	—	These bits are read as 0. The write value should be 0.	R/W
4	PLSRCSEL	PLL1 Clock Source Select 0: Main clock oscillator*3 1: HOCO*4	R/W
5	—	This bit is read as 0. The write value should be 0.	R/W
7:6	PLLMULNF[1:0]*2	PLL1 Frequency Multiplication Fractional Factor Select 0 0: 0.00 (Value after reset) 0 1: 0.33 (1/3) 1 0: 0.66 (2/3) 1 1: 0.50 (1/2)	R/W
15:8	PLLMUL[7:0]*2	PLL1 Frequency Multiplication Factor Select 0x19: × 26 (Value after reset) 0x1A: × 27 0x1B: × 28 ⋮ 0x58: × 89 0x59: ×90 0x5A: × 91 ⋮ 0xB2: ×179 0xB3: × 180 Others: Setting prohibited.	R/W

After correction

Bit	Symbol	Function	R/W
1:0	PLIDIV[1:0]*1	PLL1 Input Frequency Division Ratio Select 0 0: 1/1 0 1: 1/2 1 0: 1/3 1 1: 1/4	R/W
3:2	—	These bits are read as 0. The write value should be 0.	R/W
4	PLSRCSEL	PLL1 Clock Source Select 0: Main clock oscillator*3 1: HOCO*4	R/W
5	—	This bit is read as 0. The write value should be 0.	R/W
7:6	PLLMULNF[1:0]*2	PLL1 Frequency Multiplication Fractional Factor Select 0 0: 0.00 (Value after reset) 0 1: 0.33 (1/3) 1 0: 0.66 (2/3) 1 1: 0.50 (1/2)	R/W
15:8	PLLMUL[7:0]*2	PLL1 Frequency Multiplication Factor Select 0x19: × 26 (Value after reset) 0x34: × 53 0x35: × 54 : 0x58: × 89 0x59: ×90 0x5A: × 91 : 0xB2: ×179 0xB3: × 180 Others: Setting prohibited.	R/W

8.2.9 PLL2CCR : PLL2 Clock Control Register

[Page 211]

Before correction

Bit	Symbol	Function	R/W
1:0	PL2IDIV[1:0]*1	PLL2 Input Frequency Division Ratio Select 0 0: 1/1 (Value after reset) 0 1: 1/2 1 0: 1/3 1 1: 1/4	R/W
3:2	—	These bits are read as 0. The write value should be 0.	R/W
4	PL2SRCSEL	PLL2 Clock Source Select 0: Main clock oscillator 1: HOCO*3	R/W
5	—	These bits are read as 0. The write value should be 0.	R/W
7:6	PLL2MULNF[1:0]*2	PLL2 Frequency Multiplication Fractional Factor Select 00: 0.00 (Value after reset) 01: 0.33 (1/3) 10: 0.66 (2/3) 11: 0.50 (1/2)	R/W
15:8	PLL2MUL[7:0]*2	PLL2 Frequency Multiplication Factor Select 0x19: × 26 (Value after reset) 0x1A: × 27 0x1B: × 28 ⋮ 0x58: × 89 0x59: × 90 0x5A: × 91 ⋮ 0xD2: × 179 0xD3: × 180 Others: Setting prohibited.	R/W

RENESAS

After correction

Bit	Symbol	Function	R/W
1:0	PL2IDIV[1:0]*1	PLL2 Input Frequency Division Ratio Select 0 0: 1/1 (Value after reset) 0 1: 1/2 1 0: 1/3 1 1: 1/4	R/W
3:2	—	These bits are read as 0. The write value should be 0.	R/W
4	PL2SRCSEL	PLL2 Clock Source Select 0: Main clock oscillator 1: HOCO*3	R/W
5	—	These bits are read as 0. The write value should be 0.	R/W
7:6	PLL2MULNF[1:0]*2	PLL2 Frequency Multiplication Fractional Factor Select 00: 0.00 (Value after reset) 01: 0.33 (1/3) 10: 0.66 (2/3) 11: 0.50 (1/2)	R/W
15:8	PLL2MUL[7:0]*2	PLL2 Frequency Multiplication Factor Select 0x19: × 26 (Value after reset) 0x34: × 53 0x35: × 54 ⋮ 0x58: × 89 0x59: × 90 0x5A: × 91 ⋮ 0xD2: × 179 0xD3: × 180 Others: Setting prohibited.	R/W

8.2.16 HOCOCR : High-Speed On-Chip Oscillator Control Register

[Page 216]

Before correction

Bit	Symbol	Function	R/W
0	HCSTP	HOCO Stop 0: Operate the HOCO clock *2 1: Stop the HOCO clock	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: S-TYPE-3, P-TYPE-2

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. The HCSTP bit value after a reset is 0 when the OFS1(_SEC).HOCOEN bit is 0. It is 1 when the OFS1(_SEC).HOCOEN bit is 1.

Note 2. If you are using the HOCO (HCSTP = 0), set the OFS1(_SEC).HOCOFRQ0[2:0] bit to an optimum value.

After correction

Bit	Symbol	Function	R/W
0	HCSTP	HOCO Stop 0: Operate the HOCO clock *2 *3 1: Stop the HOCO clock	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: S-TYPE-3, P-TYPE-2

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. The HCSTP bit value after a reset is 0 when the OFS1(_SEC).HOCOEN bit is 0. It is 1 when the

OFS1(_SEC).HOCOEN bit is 1.

Note 2. If you are using the HOCO (HCSTP = 0), set the OFS1(_SEC).HOCOFRQ0[2:0] bit to an optimum value.

Note 3. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] even if OFS1(_SEC).HOCOFRQ0[2:0] is not appropriate value.

Added HOCOCR2 : High-Speed On-Chip Oscillator Control Register2

Before correction

(No description)

After correction

Base address: SYSC = 0x4001_E000

SYSC_NS = 0x5001_E000

Offset address: 0x037

Bit position:	7	6	5	4	3	2	1	0
Bit field:	–	–	–	–	–	HCFRQ0[2:0]		
Value after reset:	0	0	0	0	0	0/1*1	0/1*1	0/1*1

Bit	Symbol	Function	R/W
2:0	HCFRQ0[2:0]	HOCO Frequency Setting 0 0 0 0: 16MHz 0 0 1: 18MHz 0 1 0: 20MHz 1 0 0: 32MHz 1 1 1: 48MHz Others: Setting prohibited.	R/W
7:3	–	These bits are read as 0. The write value should be 0.	R/W

Note: S-TYPE-3, P-TYPE-2

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. Value after reset of the HCFRQ0[2:0] bits depend on OFS1(_SEC).HOCOFRQ0[2:0] bits.

The HOCOCR2 register controls the HOCO clock.

Writing to the HOCOCR2 is prohibited when the HOCOCR.HCSTP bit is 0 (the HOCO operates)

HCFRQ0[2:0] bits (HOCO Frequency Setting 0)

These bits select the frequency of HOCO.

8.2.19 FLLCR2 : FLL Control Register2

[Page 219]

Before correction

Bit	Symbol	Function	R/W
10:0 Note:	FLLCNTL[10:0] S-TYPE-3, P-TYPE-2	FLL Multiplication Control When OFS1(_SEC).HOCOFRQ0[2:0] is 000b (16 MHz) or 100b (32 MHz), these bits must be set to 0x1E9. When OFS1(_SEC).HOCOFRQ0[2:0] is 001b (18 MHz), these bits must be set to 0x226. When OFS1(_SEC).HOCOFRQ0[2:0] is 010b (20 MHz), these bits must be set to 0x263. When OFS1(_SEC).HOCOFRQ0[2:0] is 111b (48 MHz), these bits must be set to 0x1E9. Settings other than above are prohibited.	R/W
15:11	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

After correction

Bit	Symbol	Function	R/W
10:0 Note:	FLLCNTL[10:0] S-TYPE-3, P-TYPE-2	FLL Multiplication Control When OFS1(_SEC).HOCOFRQ0[2:0] is 000b (16 MHz) or 100b (32 MHz), these bits must be set to 0x1E9. When OFS1(_SEC).HOCOFRQ0[2:0] is 001b (18 MHz), these bits must be set to 0x226. When OFS1(_SEC).HOCOFRQ0[2:0] is 010b (20 MHz), these bits must be set to 0x263. When OFS1(_SEC).HOCOFRQ0[2:0] is 111b (48 MHz), these bits must be set to 0x1E9. Settings other than above are prohibited.	R/W
15:11	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note: The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.1 CPU Clock (CPUCLK)

[Page 256]

Before correction

The CPU clock (CPUCLK) is the operating clock for the CPU. Specify the frequency in the following bits:

- CPUCK[3:0] bits in SCKDIVCR2
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], and PLODIVP[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC)

After correction

The CPU clock (CPUCLK) is the operating clock for the CPU. Specify the frequency in the following bits:

- CPUCK[3:0] bits in SCKDIVCR2
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], and PLODIVP[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.2 System Clock (ICLK)

[Page 256]

Before correction

The system clock (ICLK) is the operating clock of the DMAC, DTC, Flash, SRAM, System Bus, I/O Port, and ICU. Specify the frequency in the following bits:

- ICK[3:0] bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0]its in OFS1(_SEC)

After correction

The system clock (ICLK) is the operating clock of the DMAC, DTC, Flash, SRAM, System Bus, I/O Port, and ICU. Specify the frequency in the following bits:

- ICK[3:0] bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0]its in OFS1(_SEC) [*1]

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.4 Peripheral Module Clock (PCLKA, PCLKB, PCLKC, PCLKD,PCLKE)

[Page 257]

Before correction

The peripheral module clocks (PCLKA, PCLKB, PCLKC, PCLKD and PCLKE) are the operating clocks for the peripheral modules.

The frequency of the given clock is specified in the following bits:

- PCKA[3:0], PCKB[3:0], PCKC[3:0] , PCKD[3:0] and PCKE[3:0]bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The peripheral module clocks (PCLKA, PCLKB, PCLKC, PCLKD and PCLKE) are the operating clocks for the peripheral modules.

The frequency of the given clock is specified in the following bits:

- PCKA[3:0], PCKB[3:0], PCKC[3:0] , PCKD[3:0] and PCKE[3:0]bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.5 FlashIF Clock (FCLK)

[Page 257]

Before correction

The flash interface clock (FCLK) is the operating clock for the flash memory interface. In addition to reading from the data flash, FCLK is used for the programming and erasure of the code flash and data flash.

The FCLK frequency is specified in the following bits:

● FCK[3:0] bits in SCKDIVCR

● CKSEL[2:0] bits in SCKSCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The flash interface clock (FCLK) is the operating clock for the flash memory interface. In addition to reading from the data flash, FCLK is used for the programming and erasure of the code flash and data flash.

The FCLK frequency is specified in the following bits:

● FCK[3:0] bits in SCKDIVCR

● CKSEL[2:0] bits in SCKSCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.6 External Bus Clock (BCLK, EBCLK)

[Page 258]

Before correction

The external bus clock (BCLK) is an operating clock for the external bus controller. It is also output externally from the EBCLK pin for the external connection bus.

BCLK can be output from the EBCLK pin by setting the EBCKOCR.EBCKOEN bit to 1 and setting the PmnPFS.PSEL[4:0] to 01011b. Make sure that modification of the PmnPFS.PSEL[4:0] to 01011b must always be performed while the EBCKOCR.EBCKOEN bit is 0.

When the BCKCR.BCLKDIV bit is set to 1, the BCLK clock divided by 2 is output from the EBCLK pin.

Specify the frequency in the following bits:

- BCK[3:0] bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The external bus clock (BCLK) is an operating clock for the external bus controller. It is also output externally from the EBCLK pin for the external connection bus.

BCLK can be output from the EBCLK pin by setting the EBCKOCR.EBCKOEN bit to 1 and setting the PmnPFS.PSEL[4:0] to 01011b. Make sure that modification of the PmnPFS.PSEL[4:0] to 01011b must always be performed while the EBCKOCR.EBCKOEN bit is 0.

When the BCKCR.BCLKDIV bit is set to 1, the BCLK clock divided by 2 is output from the EBCLK pin.

Specify the frequency in the following bits:

- BCK[3:0] bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) [*1]

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.7 SDRAM Clock (SDCLK)

[Page 258]

Before correction

The SDRAM clock (SDCLK) is an operating clock for the external bus controller. It is output externally from the SDCLK pin for the SDRAM that is connected to the external bus. To output SDCLK on the SDCLK pin, set the SDCKOCR.SDCKOEN bit to 1 and set the PmnPFS.PSEL[4:0] bits to 01011b(enabling SDCLK output). Only change the value in the PmnPFS.PSEL[4:0] bits when the SDCKOCR.SDCKOEN bit is 0. Specify the frequency in the following bits:

- BCK[3:0] bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The SDRAM clock (SDCLK) is an operating clock for the external bus controller. It is output externally from the SDCLK pin for the SDRAM that is connected to the external bus. To output SDCLK on the SDCLK pin, set the SDCKOCR.SDCKOEN bit to 1 and set the PmnPFS.PSEL[4:0] bits to 01011b(enabling SDCLK output). Only change the value in the PmnPFS.PSEL[4:0] bits when the SDCKOCR.SDCKOEN bit is 0. Specify the frequency in the following bits:

- BCK[3:0] bits in SCKDIVCR
- CKSEL[2:0] bits in SCKSCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.9 SCI Clock (SCICLK)

[Page 258]

Before correction

The SCI clock (SCICLK) is the operating clock for the SCI module.

Specify the frequency in the following bits:

- SCICKDIV[2:0] bits in SCICKDIVCR
- SCICKSEL[3:0] bits in SCICKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The SCI clock (SCICLK) is the operating clock for the SCI module.

Specify the frequency in the following bits:

- SCICKDIV[2:0] bits in SCICKDIVCR
- SCICKSEL[3:0] bits in SCICKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.10 SPI Clock (SPICLK)

[Page 259]

Before correction

The SPI clock (SPICLK) is the operating clock for the SPI module.

Specify the frequency in the following bits:

- SPICKDIV[2:0] bits in SPICKDIVCR
- SPICKSEL[3:0] bits in SPICKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The SPI clock (SPICLK) is the operating clock for the SPI module.

Specify the frequency in the following bits:

- SPICKDIV[2:0] bits in SPICKDIVCR
- SPICKSEL[3:0] bits in SPICKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) [*1]

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.11 Octal-SPI clock (OCTACLK, OCTADIVCLK)

[Page 259]

Before correction

The Octal-SPI clock (OCTACLK) is the operating clock for the Octal-SPI module.

Specify the frequency in the following bits:

- OCTACKDIV[2:0] bits in OCTACKDIVCR
- OCTACKSEL[3:0] bits in OCTACKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The Octal-SPI clock (OCTACLK) is the operating clock for the Octal-SPI module.

Specify the frequency in the following bits:

- OCTACKDIV[2:0] bits in OCTACKDIVCR

● OCTACKSEL[3:0] bits in OCTACKCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.12 CANFD Core clock (CANFDCLK)

[Page 259]

Before correction

The CANFD Core clock (CANFDCLK) is the operating clock for the CANFD module.

Specify the frequency in the following bits:

● CANFDCKDIV[2:0] bits in CANFDCKDIVCR

● CANFDCKSEL[3:0] bits in CANFDCKCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The CANFD Core clock (CANFDCLK) is the operating clock for the CANFD module.

Specify the frequency in the following bits:

● CANFDCKDIV[2:0] bits in CANFDCKDIVCR

● CANFDCKSEL[3:0] bits in CANFDCKCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.13 USB Clock (USBCLK)

[Page 259]

Before correction

The USB clock (USBCLK) is the operating clock for the USBFS and USBHS module.

A 48-MHz clock must be supplied when using the USBFS module or when using the USBHS module in CL-Only mode.

USBCLK does not need to be supplied when not using USBHS module in CL-Only mode.

The USBCLK frequency is specified in the following bits:

● USBCKDIV[2:0] bits in USBCKDIVCR

● USBCKSEL[3:0] bits in USBCKCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC).

<u>After correction</u>

The USB clock (USBCLK) is the operating clock for the USBFS and USBHS module.

A 48-MHz clock must be supplied when using the USBFS module or when using the USBHS module in CL-Only mode.

USBCLK does not need to be supplied when not using USBHS module in CL-Only mode.

The USBCLK frequency is specified in the following bits:

● USBCKDIV[2:0] bits in USBCKDIVCR

● USBCKSEL[3:0] bits in USBCKCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.14 USB Clock (USB60CLK)

[Page 260]

<u>Before correction</u>

The USB clock (USB60CLK) is the operating clock for the USBHS module. A 60-MHz clock must be supplied when using the USBHS module in CL-Only mode. USB60CLK does not need to be supplied when not using USBHS in CL-Only mode.

Specify the frequency in the following bits:

● USB60CKDIV[2:0] bits in USB60CKDIVCR

● USB60CKSEL[3:0] bits in USB60CKCR

● PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2

● PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2

● HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The USB clock (USB60CLK) is the operating clock for the USBHS module. A 60-MHz clock must be supplied when using the USBHS module in CL-Only mode. USB60CLK does not need to be supplied when not using USBHS in CL-Only mode.

Specify the frequency in the following bits:

- USB60CKDIV[2:0] bits in USB60CKDIVCR
- USB60CKSEL[3:0] bits in USB60CKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.16 LCD Clock (LCDCLK)

[Page 260]

Before correction

The LCD clock (LCDCLK) is the operating clock for the LCD module.

Specify the frequency in the following bits:

- LCDCKDIV[2:0] bits in LCDCLDIVCR
- LCDCKSEL[3:0] bits in LCDCKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The LCD clock (LCDCLK) is the operating clock for the LCD module.

Specify the frequency in the following bits:

- LCDCKDIV[2:0] bits in LCDCLDIVCR
- LCDCKSEL[3:0] bits in LCDCKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC). *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.17 I3C Clock (I3CCLK)

[Page 260]

Before correction

The I3C clock (I3CCLK) is the operating clock for the I3C module.

Specify the frequency in the following bits:

- I3CCKDIV[2:0] bits in I3CCKDIVCR
- I3CCKSEL[3:0] bits in I3CCKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC).

After correction

The I3C clock (I3CCLK) is the operating clock for the I3C module.

Specify the frequency in the following bits:

- I3CCKDIV[2:0] bits in I3CCKDIVCR
- I3CCKSEL[3:0] bits in I3CCKCR
- PLLMUL[7:0], PLLMULNF[1:0], PLIDIV[1:0], PLODIVP[3:0], PLODIVQ[3:0] and PLODIVR[3:0] bits in PLLCCR and PLLCCR2
- PLL2MUL[7:0], PLL2MULNF[1:0], PL2IDIV[1:0], PL2ODIVP[3:0], PL2ODIVQ[3:0] and PL2ODIVR[3:0] bits in PLL2CCR and PLL2CCR2
- HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

8.10.26 External Pin Output Clock (CLKOUT)

[Page 261]

Before correction

The CLKOUT is output externally from the CLKOUT pin for the clock or buzzer output. The CLKOUT is output to the CLKOUT pin when the CKOCR.CKOEN bit is set to 1. Only change the value in the CKODIV[2:0] bits or CKOSEL[2:0] bits in CKOCR when the CKOCR.CKOEN bit is 0.

The CLKOUT clock frequency is specified in the following bits:

- CKODIV[2:0] bits or CKOSEL[2:0] bits in CKOCR
- HOCOFRQ0[2:0] bits in OFS1(_SEC)

After correction

The CLKOUT is output externally from the CLKOUT pin for the clock or buzzer output. The CLKOUT is output to the CLKOUT pin when the CKOCR.CKOEN bit is set to 1. Only change the value in the CKODIV[2:0] bits or CKOSEL[2:0] bits in CKOCR when the CKOCR.CKOEN bit is 0.

The CLKOUT clock frequency is specified in the following bits:

● CKODIV[2:0] bits or CKOSEL[2:0] bits in CKOCR

● HOCOFRQ0[2:0] bits in OFS1(_SEC) *1

Note1. OFS1 is for non-secure developers and OFS1_SEC is for secure developers. The applied setting value is determined by OFS1_SEL. The value of OFS1(_SEC).HOCOFRQ0[2:0] bits is automatically transferred to HOCOCR2.HCFRQ0[2:0] bits after reset, therefore HOCO frequency can also be specified by HOCOCR2.HCFRQ0[2:0] bits.

Table 8.14 Example of the HOCO initial setting procedure after reset release / after Deep Software Standby cancellation (OFS1(_SEC).HOCOEN = 1, without FLL)

[Page 270]

Before correction

No.	Step	Description
1	Start	The HOCO is stopped after reset release / Deep software standby cancellation when OFS1(_SEC).HOCOEN is 1.
2	Cancel Register Write Protection	Set 1 to PRC0 bit and PRC1 bit in PRCR register
3	Check the HOCO power supply*1	Check the following bit in the HOCOLDOCR register ● LDOSTP bit is 0 (LDO is enabled)
4	Set the oscillation keep in Software Standby mode	If HOCO keeps oscillation in Software Standby mode, set the following: ● HOCOSOKP bit in HOCOSCR register ● SKEEP bit in HOCOLDOCR register
5	Set HOCO to operate	Set HOCO to start oscillating with the HOCOCR register.
6	Wait for HOCO clock oscillation to stabilize	Polling until HOCOSF bit in OSCSF register is read as 1 (Oscillation is stable)
7	Apply Register Write Protection	Set 0 to PRC0 bit and PRC1 bit in PRCR register
8	End	HOCO clock setting is completed. HOCO Clock is available.

After correction

No.	Step	Description
1	Start	The HOCO is stopped after reset release / Deep software standby cancellation when OFS1(_SEC).HOCOEN is 1.
2	Cancel Register Write Protection	Set 1 to PRC0 bit and PRC1 bit in PRCR register
3	Check the HOCO power supply*1	Check the following bit in the HOCOLDOCR register ● LDOSTP bit is 0 (LDO is enabled)
4	Set the HOCO frequency	Set the HOCO frequency with HOCOCR2 register.
5	Set the oscillation keep in Software Standby mode	If HOCO keeps oscillation in Software Standby mode, set the following: ● HOCOSOKP bit in HOCOSCR register ● SKEEP bit in HOCOLDOCR register
6	Set HOCO to operate	Set HOCO to start oscillating with the HOCOCR register.
7	Wait for HOCO clock oscillation to stabilize	Polling until HOCOSF bit in OSCSF register is read as 1 (Oscillation is stable)
8	Apply Register Write Protection	Set 0 to PRC0 bit and PRC1 bit in PRCR register
9	End	HOCO clock setting is completed. HOCO Clock is available.

Table 8.15 Example of the HOCO setting procedure with FLL function after reset release / after Deep Software

Standby cancellation (OFS1(_SEC).HOCOEN = 1)

[Page 270, 271]

Before correction

No.	Step	Description
1	Start	The HOCO is stopped after reset release / Deep software standby cancellation when OFS1(_SEC).HOCOEN is 1.
2	Cancel Register Write Protection	Set 1 to PRC0 bit and PRC1 bit in PRCR register

No.	Step	Description
3	Check the HOCO power supply*1	Check the following bit in the HOCOLDOCR register. ● LDOSTP bit is 0 (LDO is enabled)
4	Set the FLL function to enable*2	Set the FLL Multiplication Control with FLLCR2 register. Set the FLL function to enable with FLLCR1 register.
5	Set HOCO to operate	Set HOCO to start oscillating with the HOCOCR register.
6	Wait for HOCO clock oscillation to stabilize	Polling until HOCOSF bit in OSCSF register is read as 1 (Oscillation is stable)
7	Wait for FLL stabilization	Wait for FLL stabilization wait time (t_{FLLWT}), or wait until the HOCO clock is measured to confirm that the frequency accuracy is stable.
8	Check the HOCO stabilization	Check that HOCOSF bit in OSCSF register is read as 1
9	Apply Register Write Protection	Set 0 to PRC0 bit and PRC1 bit in PRCR register
10	End	FLL setting is completed. HOCO Clock is available.

After correction

No.	Step	Description
1	Start	The HOCO is stopped after reset release / Deep software standby cancellation when OFS1(_SEC).HOCOEN is 1.
2	Cancel Register Write Protection	Set 1 to PRC0 bit and PRC1 bit in PRCR register

No.	Step	Description
3	Check the HOCO power supply*1	Check the following bit in the HOCOLDOCR register. ● LDOSTP bit is 0 (LDO is enabled)
4	Set the HOCO frequency	Set the HOCO frequency with HOCOCR2 register.
5	Set the FLL function to enable*2	Set the FLL Multiplication Control with FLLCR2 register. Set the FLL function to enable with FLLCR1 register.
6	Set HOCO to operate	Set HOCO to start oscillating with the HOCOCR register.
7	Wait for HOCO clock oscillation to stabilize	Polling until HOCOSF bit in OSCSF register is read as 1 (Oscillation is stable)
8	Wait for FLL stabilization	Wait for FLL stabilization wait time (t_{FLLWT}), or wait until the HOCO clock is measured to confirm that the frequency accuracy is stable.
9	Check the HOCO stabilization	Check that HOCOSF bit in OSCSF register is read as 1
10	Apply Register Write Protection	Set 0 to PRC0 bit and PRC1 bit in PRCR register
11	End	FLL setting is completed. HOCO Clock is available.

10. Low Power Modes

Table 10.3 Operating state of each low power mode

[Page 287]

Before correction

Note 9. For the address bus and bus control signals (For SRAM : [CS0 to CS7, RD, WR0 to WR1, WR, BC0 to BC1 and ALE], and for SDCS, RAS, CAS and WE]), keeping the output state or changing to the high-impedance state can be selected by SBYCR.OPE bit.

After correction

Note 9. For the address bus and bus control signals (For SRAM: [A00 to A23, CS0 to CS7, RD, WR0 to WR3, WR, BC0 to BC3 and ALE], and for SDRAM: [A00 to A16, DQM0 to DQM3, SDCS, RAS, CAS, WE and CKE]), keeping the output state or changing to the high-impedance state can be selected by SBYCR.OPE bit.

10.2.10 SBYCR : Standby Control Register

[Page 303]

Before correction

OPE bit (Output Port Enable)

The OPE bit specifies whether to set to the high-impedance state or to retain the output of the address bus and bus control signals. For SRAM : [(CS0 to CS7, RD, WR0 to WR1, WR, BC0 to BC1, and ALE) in Software Standby mode or Deep Software Standby mode.

After correction

OPE bit (Output Port Enable)

The OPE bit specifies whether to set to the high-impedance state or to retain the output of the address bus and bus control signals. For SRAM: A00 to A23, CS0 to CS7, RD, WR0 to WR3, WR, BC0 to BC3 and ALE, and for SDRAM: A00 to A16, DQM0 to DQM3, SDCS, RAS, CAS, WE and CKE in Software Standby mode or Deep Software Standby mode.

12. Register Write Protection

Table 12.1 Association between the bits in the PRCR register and registers to be protected

[Page 361]

Before correction

PRCR bit	Register to be protected
PRC0	● Registers related to the clock generation circuit:
PRC1	● Registers related to the low power modes: SBYCR, OPCCR, PDCTRGD, PDRAMSCR0, PDRAMSCR1, SSCR1, LPSCR, DPSBYCR, DPSWCR, DPSIER0-3, DPSIFR0-3, DPSIEGR0-2, PLL1LDOCR, PLL2LDOCR, HOCOLDOCR, LVOCR ● Register related to the battery backup function: VBTBER, VBTICTLR, VBTBKR[n] (n = 0 to 127), VBTBPCR1, VBTBPCR2, VBTBPSR, VBTADSR, VBTADCR1, VBTADCR2, VBTICTLR2
PRC3	● Registers related the PVD: PVD1CR1, PVD1SR, PVD2CR1, PVD2SR, PVD1CMPCR, PVD2CMPCR, PVD1CR0, PVD2CR0, PVD1FCR, PVD2FCR, VBATTMNSELR
PRC4	● Registers related to the Security and Privilege setting registers: ELCSARx (x=A,B)*1, ELCPARx (x=A,B), PSARx (x=A to E), MSSAR, PPARx (x=A to E), MSPAR, PmSAR (m=0 to 9, A to G), CPUSAR, DEBUGSAR, ICUSARx (x=A,B,E to I), SRAMSAR, BUSSARx(x=A to C), BUSPARC, MMPUSARx (x=A,B), DTCSAR, DMACSAR, DMACCHSAR, DMACCHPAR, TEVTRCR, SRAMSABAR0-1, STBRAMSABAR, STBRAMPABAR_NS, STBRAMPABAR_S, FSAR, CGFSAR, RSTSAR, LPMSAR, PVDSAR, BBFSAR, DPFSAR, RSCSAR, PGCSAR, VBRSABAR, VBRPABARS, VBRPABARNS
PRC5*1	● Registers related to the reset control SYRSTMSK0, SYRSTMSK2

Note 1. Only PRCR_S is supported.

After correction

PRCR bit	Register to be protected
PRC0	● Registers related to the clock generation circuit: SCKDIVCR, SCKDIVCR2, SCKSCR, PLLCCR, PLLCR, BCKCR, MOSCCR, HOCOCR, HOCOCR2, MOCOCR, FLLCR1, FLLCR2, CKOCR, OSTDCR, OSTDSR, PLL2CCR, PLL2CR, PLLCCR2, PLL2CCR2, EBCKOCR, SDCKOCR, SCICKDIVCR, SCICKCR, SPICKDIVCR, SPICKCR, LCDCKDIVCR, LCDCKCR, MOCOUTCR, HOCOUTCR, USBCKDIVCR, OCTACKDIVCR, CANFDCKDIVCR, USB60CKDIVCR, I3CCKDIVCR, USBCKCR, OCTACKCR, CANFDCKCR, USB60CKCR, I3CCKCR, MOSCSCR, HOCOSCR, MOSCWTCR, MOMCR, SOSCCR, SOMCR, LOCOCR, LOCOUTCR, SYRACCR
PRC1	● Registers related to the low power modes: SBYCR, OPCCR, PDCTRGD, PDRAMSCR0, PDRAMSCR1, SSCR1, LPSCR, DPSBYCR, DPSWCR, DPSIER0-3, DPSIFR0-3, DPSIEGR0-2, PLL1LDOCR, PLL2LDOCR, HOCOLDOCR, LVOCR ● Register related to the battery backup function: VBTBER, VBTICTLR, VBTBKR[n] (n = 0 to 127), VBTBPCR1, VBTBPCR2, VBTBPSR, VBTADSR, VBTADCR1, VBTADCR2, VBTICTLR2
PRC3	● Registers related the PVD: PVD1CR1, PVD1SR, PVD2CR1, PVD2SR, PVD1CMPCR, PVD2CMPCR, PVD1CR0, PVD2CR0, PVD1FCR, PVD2FCR, VBATTMNSELR
PRC4	● Registers related to the Security and Privilege setting registers: ELCSARx (x=A,B)[*1], ELCPARx (x=A,B), PSARx (x=B to E), MSSAR, PPARx (x=B to E), MSPAR, PmSAR (m=0 to 9, A, B), CPUSAR, DEBUGSAR, ICUSARx (x=A,B,E to I), SRAMSAR, BUSSARx(x=A to C), BUSPARC, MMPUSARx (x=A,B), DTCSAR, DMACSAR, DMACCHSAR, DMACCHPAR, TEVTRCR, SRAMSABAR0-1, STBRAMSABAR, STBRAMPABAR_NS, STBRAMPABAR_S, FSAR, CGFSAR, RSTSAR, LPMSAR, PVDSAR, BBFSAR, DPFSAR, RSCSAR, PGCSAR, VBRSABAR, VBRPABARS, VBRPABARNS
PRC5[*1]	● Registers related to the reset control SYRSTMSK0, SYRSTMSK2

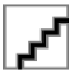
Note 1. Only PRCR_S is supported.

14. Buses

Figure 14.63 Timing example of consecutive read (2)

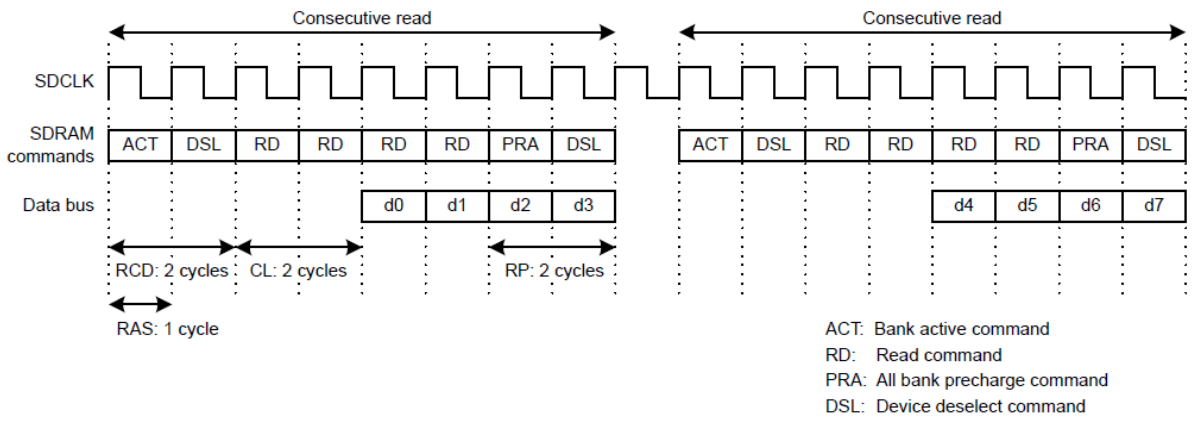
[Page 499]

Before correction



(wrong figure)

After correction



23. Ultra-Low-Power Timer (ULPT)

23.4.7 Standby mode

[Page 879]

Before correction

It is prohibited to rewrite the ULPT, ULPTCMA, and ULPTCMB registers immediately before setting each standby mode.

If the ULPT, ULPTCMA, and ULPTCMB registers are rewritten while the counter is running, set each standby mode after four or more cycles of the count source.

After correction

It is prohibited to rewrite the ULPTCNT, ULPTCMA, and ULPTCMB registers immediately before setting each standby mode.

If the ULPTCNT, ULPTCMA, and ULPTCMB registers are rewritten while the counter is running, set each standby mode after four or more cycles of the count source.

23.5.9 Restrictions on ULPTEEx-DS and ULPTEVIx-DS pins at DSTBY1

[Page 881]

Before correction

When using the ULPTEEx-DS and ULPTEVIx-DS pins in DSTBY1, set the input pins to 0 at the time of entering DSBTY1. After the DSTBY1 transition, these input pins can be used for pulse input (both 0 and 1 input is allowed). In order to see the transition to DSTBY1, IO port can be used utilizing the change when deep software standby happens.

(Examples of flow)

1. Use GPIO outputs to transmit to external devices before executing WFI instructions.
2. Execute WFI instruction and move to DSTBY1.
3. The external device detects the GPIO output. After 1ms seconds have passed, input is started for the ULPT.

After correction

If external events are to be counted continuously during Deep Software Standby mode 1, the ULPTEEx-DS and ULPTEVIx-DS pins should be held low during the transition to Deep Software Standby mode 1.

The device that generates the external events must be notified of the transition to Deep Software Standby mode 1 using the general I/O port.

An example of the procedure is shown below. (Examples of flow)

(a) Procedure on the MCU

1. During initial setup, set the port for notifying the external device as a general output port and output low.
2. Before transitioning to Deep Software Standby mode 1, set the PODR bit of the above port to 1.
3. Read the PODR bit and confirm that it is set to 1.
4. After 10 μs has passed, execute the WFI instruction.

(b) Procedure on the External Device

1. When the above port goes high, hold the ULPTEEx-DS and ULPTEVIx-DS pins low within 10 μs. If the time exceeds 10 μs, add the excess time to the time in 4 above. If the time is less than 10 μs, subtract the reduced time.
2. After 1 ms or more has passed, resume the event output.

32. I2C Bus Interface (IIC)

32.8 Wakeup Function

[Page 1457]

Before correction

Precautions on the use of the wakeup function

- Do not change the content of the IIC registers except the WUSEN bit in ICWUR2 while the WUASYF flag in ICWUR2 is 1 (during PCLKB asynchronous operation).
- Set ICWUR.WUE and ICWUR.WUIE to 1, and ICCR2.MST and ICCR2.TRS to 0 (slave reception mode) before switching to PCLKB asynchronous mode.
- The device ID and the 10-bit slave address cannot be selected for the wakeup interrupt source. Set the DIDE bit in ICSER and FS bit in SARUy (y = 0 to 2) to 0.
- Set bits TIE, TEIE, RIE, NAKIE, SPIE, STIE, ALIE, and TMOIE in the ICIER register to 0 (interrupt disabled) before switching to the asynchronous operation.
- When the wakeup function is enabled, do not use the timeout function (ICWUR.WUE = 1)
- Even when a wakeup interrupt is generated during PCLKB asynchronous operation (when ICWUR2.WUASYF = 1), if the slave addresses match in PCLKB synchronous mode (ICWUR2.WUASYF = 0), the wakeup interrupt does not occur and the WUF flag is not set.
- If the timing of writing 0 to the ICWUR2.WUSEN bit and the timing of detecting a start condition conflict, the IIC might start the next reception in PCLKB synchronous operation mode. In this case, ICWUR2.WUASYF flag becomes 1 (switch to PCLKB asynchronous mode) when data communication is complete, a stop condition is detected, and detection of a wakeup event starts.
- If you want to switch from PCLKB asynchronous operation to PCLKB synchronous operation without address match detection, it will switch in the stop condition detection. When the ICWUR2.WUSEN bit was set to 1 in a bus free state, it is continued PCLKB asynchronous operation (Reception operation: waiting communication frame). ICWUR2.WUSYF flag becomes to 1 when IIC detect the stop condition of the next communication frame, and IIC switches to PCLKB synchronous operation.
- After writing 0 to the WUSEN bit in ICWUR2, do not change registers relate to the IIC operation mode setting (ICMR3, ICSER, and SARLy) until the mode is switched to PCLKB asynchronous operation from PCLKB synchronous operation (while the ICWUR2.WUASYF flag is 1). If the register value changes during this period by an interrupt handling or another factor, the IIC might malfunction before switching to the asynchronous operation.
- During PCLKB asynchronous operation (ICWUR2.WUSYF = 0 (or WUASYF = 1)), do not refer to each flag of ICSR1, ICSR2 register and ICCR2.BBSY flag.

After correction

Precautions on the use of the wakeup function

- Do not change the content of the IIC registers except the ICIER register and WUSEN bit in ICWUR2 while the WUASYF flag in ICWUR2 is 1 (during PCLKB asynchronous operation).
- Set ICWUR.WUE and ICWUR.WUIE to 1, and ICCR2.MST and ICCR2.TRS to 0 (slave reception mode) before

switching to PCLKB asynchronous mode.

- The device ID and the 10-bit slave address cannot be selected for the wakeup interrupt source. Set the DIDE bit in ICSER and FS bit in SARUy (y = 0 to 2) to 0.
- Set bits TIE, TEIE, RIE, NAKIE, SPIE, STIE, ALIE, and TMOIE in the ICIER register to 0 (interrupt disabled) after switching to PCLKB asynchronous operation (ICWUR2.WUASYF=1).
- When the wakeup function is enabled, do not use the timeout function (ICWUR.WUE = 1)
- Even when a wakeup interrupt is generated during PCLKB asynchronous operation (when ICWUR2.WUASYF = 1), if the slave addresses match in PCLKB synchronous mode (ICWUR2.WUASYF = 0), the wakeup interrupt does not occur and the WUF flag is not set.
- If the timing of writing 0 to the ICWUR2.WUSEN bit and the timing of detecting a start condition conflict, the IIC might start the next reception in PCLKB synchronous operation mode. In this case, ICWUR2.WUASYF flag becomes 1 (switch to PCLKB asynchronous mode) when data communication is complete, a stop condition is detected, and detection of a wakeup event starts.
- If you want to switch from PCLKB asynchronous operation to PCLKB synchronous operation without address match detection, it will switch in the stop condition detection. When the ICWUR2.WUSEN bit was set to 1 in a bus free state, it is continued PCLKB asynchronous operation (Reception operation: waiting communication frame). ICWUR2.WUSYF flag becomes to 1 when IIC detect the stop condition of the next communication frame, and IIC switches to PCLKB synchronous operation.
- After writing 0 to the WUSEN bit in ICWUR2, do not change registers relate to the IIC operation mode setting (ICMR3, ICSER, and SARLy) until the mode is switched to PCLKB asynchronous operation from PCLKB synchronous operation (while the ICWUR2.WUASYF flag is 1). If the register value changes during this period by an interrupt handling or another factor, the IIC might malfunction before switching to the asynchronous operation.
- During PCLKB asynchronous operation (ICWUR2.WUSYF = 0 (or WUASYF = 1)), do not refer to each flag of

ICSR1, ICSR2 register and ICCR2.BBSY flag.

34. CAN with Flexible Data-rate (CANFD)

34.2.56 CFDGLOCKK : Global Lock Key Register

[Page 1808]

Before correction

LOCK[15:0] bits (Lock Key)

The unlock key sequence must be written in the LOCK[15:0] bits to configure the CANFD module in FIFO OTB disable and RAM test modes.

The read value from these bits is always 0x0000.

You cannot write to these bits when the CANFD module is in GL_SLEEP or GL_RESET mode.

Do not write to these bits when the CANFD module is in GL_OPERATION mode.

After correction

LOCK[15:0] bits (Lock Key)

The unlock key sequence must be written in the LOCK[15:0] bits to configure the CANFD module in RAM test modes.

The read value from these bits is always 0x0000.

You cannot write to these bits when the CANFD module is in GL_SLEEP or GL_RESET mode.

Do not write to these bits when the CANFD module is in GL_OPERATION mode.

34.6.2.1 FIFO Buffers Configuration

[Page 1868, 1869]

Before correction

(3) FIFO depth configuration

(omission)

The RAM allocation for RX message buffers along with FIFO buffers is limited to 16 messages with 64 data bytes.

Configuration of the RX message buffers, along with FIFO buffers, that exceeds this maximum limit should not be done.

(omission)

(4) FIFO payload size configuration

(omission)

The RAM allocation for RX message buffers along with FIFO buffers is limited to 16 messages with 64 data bytes.

Configuration of the RX message buffers, along with FIFO buffers, that exceeds this maximum limit should not be done.

After correction

(3) FIFO depth configuration

(omission)

The RAM allocation for RX message buffers along with FIFO buffers is limited to 16 messages with 64 data bytes (76 bytes including ID and PTR).

Configuration of the RX message buffers, along with FIFO buffers, that exceeds this maximum limit should not be done.

(omission)

(4) FIFO payload size configuration

(omission)

The RAM allocation for RX message buffers along with FIFO buffers is limited to 16 messages with 64 data bytes (76 bytes including ID and PTR).

Configuration of the RX message buffers, along with FIFO buffers, that exceeds this maximum limit should not be done.

34.9.2.1 RAM Test Mode

[Page 1900]

Before correction

- MB RAM:

  pn = ceil (2072 / 256) = 9 pages

CFDGTSTCFG.RTMPS[3:0] = 0 to 8 inclusive

After correction

● MB RAM:

pn = ceil (2072 / 256) = 9 pages

CFDGTSTCFG.RTMPS[3:0] = 0 to 8 inclusive

(User should not access more than 24 Bytes in the last page)

Added 34.10 RAM area configuration

Before correction

(no description)

After correction

The RAM area used in CANFD (referred to as MRAM) can be split into the following groups as shown below in Figure 34.54:

AFL Rule Table area
PFL Rule Table area
Message Buffer[*1] area (RX MB + FIFO Buffer)
OTB area
THL area
TX MB area

Physically the RAM is the Message Buffer RAM[*2] (RX MB, RX FIFO, Common FIFO[*3,] TX MB, THL, OTB, AFL Rule Table, PFL Rule Table).

*1: Referred to as MB
*2: Referred to as MRAM
*3: Referred to as CFIFO

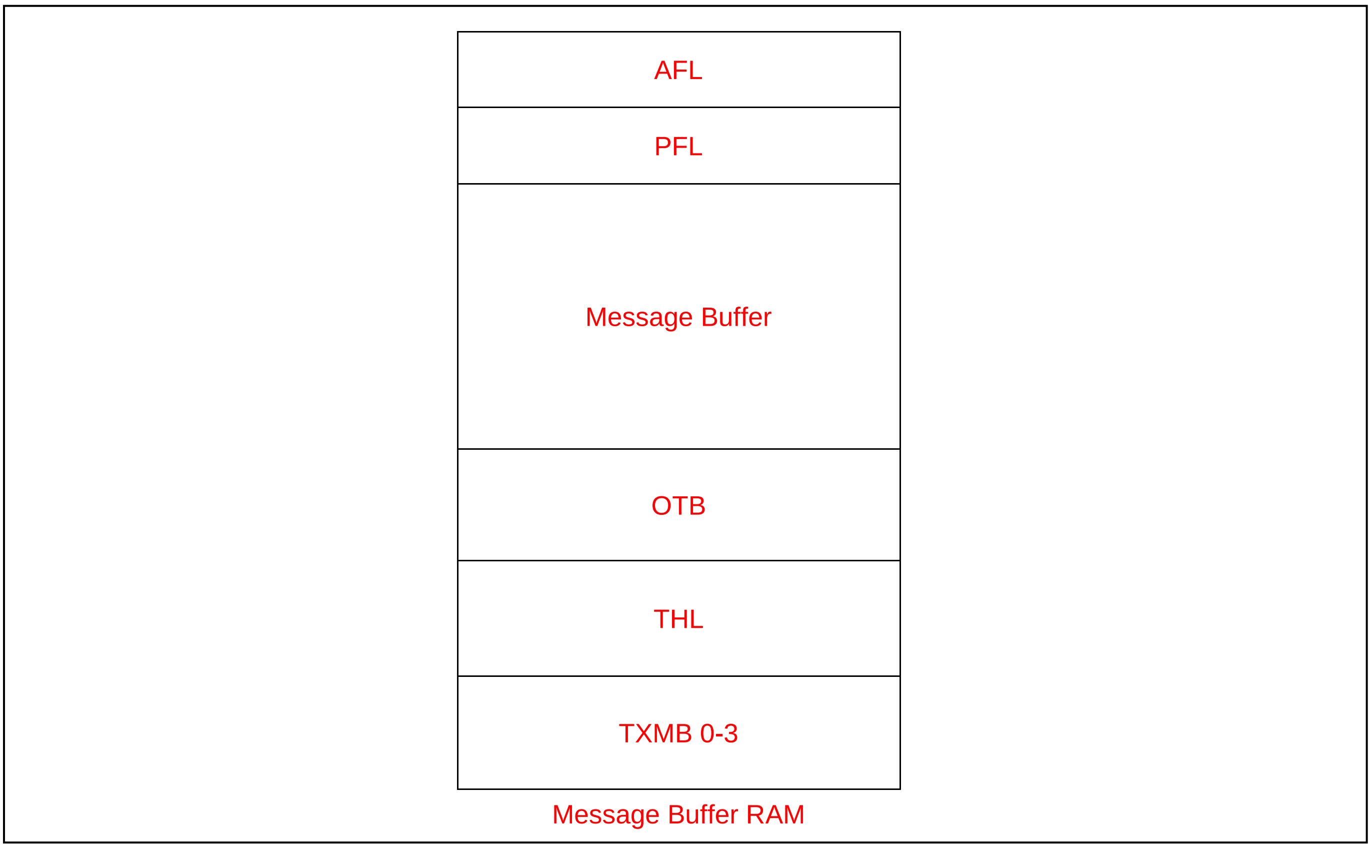


Figure 34.54 RAM area grouping

The MRAM area starts with the TX MB area at address 0x0000. The TX MB area is followed immediately by the THL area which is then followed immediately by the OTB area. The size of the TX MB, THL and OTB area is fixed.
The OTB area is followed by the message buffer area. The message buffer area size depends on the configuration of the flat RXMBs, RXFIFOs and CFIFO. When all are configured the RX MB area is followed by the RXFIFO area which is followed by the CFIFO area.

The configured MRAM area can be calculated then as follows.

MRAM_cfg = RXMB_MRAM_cfg + RXFIFO_MRAM_cfg + CFIFO_MRAM_cfg +
TXMB_MRAM_cfg +THL_MRAM_cfg + OTB_MRAM_cfg +
AFL_MRAM_cfg + PFL_MRAM_cfg

RXMB_MRAM_cfg = (12 Bytes + CFDRMNB.RMPLS) * CFDRMNB.NRXMB
RXFIFO_MRAM_cfg = SUM((12 Bytes + CFDRFCCa.RFPLS) * CFDRFCCa.RFDC)
CFIFO_MRAM_cfg = (12 Bytes + CFDCFCC.CFPLS) * CFDCFCC.CFDC
TXMB_MRAM_cfg = 304 Bytes
THL_MRAM_cfg = 64 Bytes
OTB_MRAM_cfg = 160 Bytes
PFL_MRAM_cfg = 72 Bytes
AFL_MRAM_cfg = 256 Bytes

“a” means RX FIFO index = [0…no_of_RFIFOs-1]
no_of_RFIFOs : Number of configured RX FIFOs

Note: For CFDRFCCa.RFDC, CFDCFCC.CFDC, CFDRMNB.RMPLS, CFDRMNB.NRXMB, CFDRFCCa.RFPLS and CFDCFCC.CFPLS the related number of bytes must be used.

The Table 34.29 shows the calculation of the different RAM areas used for the AFL entries, OTB buffers, TX/RX message buffers, RX/Common FIFOs and PFL entries.

Table 34.29 MRAM area calculation

RAM Name	RAM Property	RAM Area Calculation Method	RAM Values
AFL	Avg. rule entries		16
	No. of Bytes in a rule entry	Fixed	16
	No. of Bytes in AFL area	Avg. rule entries * No. of Bytes in a rule entry	256
PFL	Avg. rule entries		2
	No. of Bytes in a rule entry	Fixed	36
	No. of Bytes in PFL area	Avg. rule entries * No. of Bytes in a rule entry	72
TX MB	No. of TX MBs	Fixed	4
	No. of Bytes needed for each TX MB	Fixed	76
	No. of Bytes in TX MB area	No. of TXMBs * No. of Bytes needed for each TXMB	304
THL	No. of entries in 1 THL buffer	Fixed	8
	No. of Bytes needed for each THL entry	Fixed	8
	No. of Bytes in THL area	No. of entries in 1 THL buffer * No. of Bytes needed for each THL entry	64
OTB	Avg. No. of buffers		2
	No. of Bytes for OTB entry	Fixed	80
	No.of Bytes in OTB area	Avg. No. of buffers * No. of Bytes for OTB entry	160
Message Buffer	No. of RXMBs	Fixed	16
	No. of RXFIFOs	Fixed	2
	No. of Common FIFO	Fixed	1
	Avg. No. of messages for RXMB and FIFO buffers		16
	No. of Bytes for each stored message	Fixed	-
	Average size of a Message Buffer in Bytes		76
	No. of Bytes in Message Pool area	Avg. No. of messages for RXMB and FIFO buffers * Average size of a Message Buffer in Bytes	1216
	No. of Bytes Message RAM	No. of Bytes in Message Pool area + No. of Bytes in OTB area + No. of Bytes in THL area + No. of Bytes in TXMB area + No. of Bytes in PFL area + No. of Bytes in AFL area	2072

Added 34.10.1 Examples

Before correction

(no description)

After correction

RENESAS

The Figure 34.55 below shows one possible configuration.

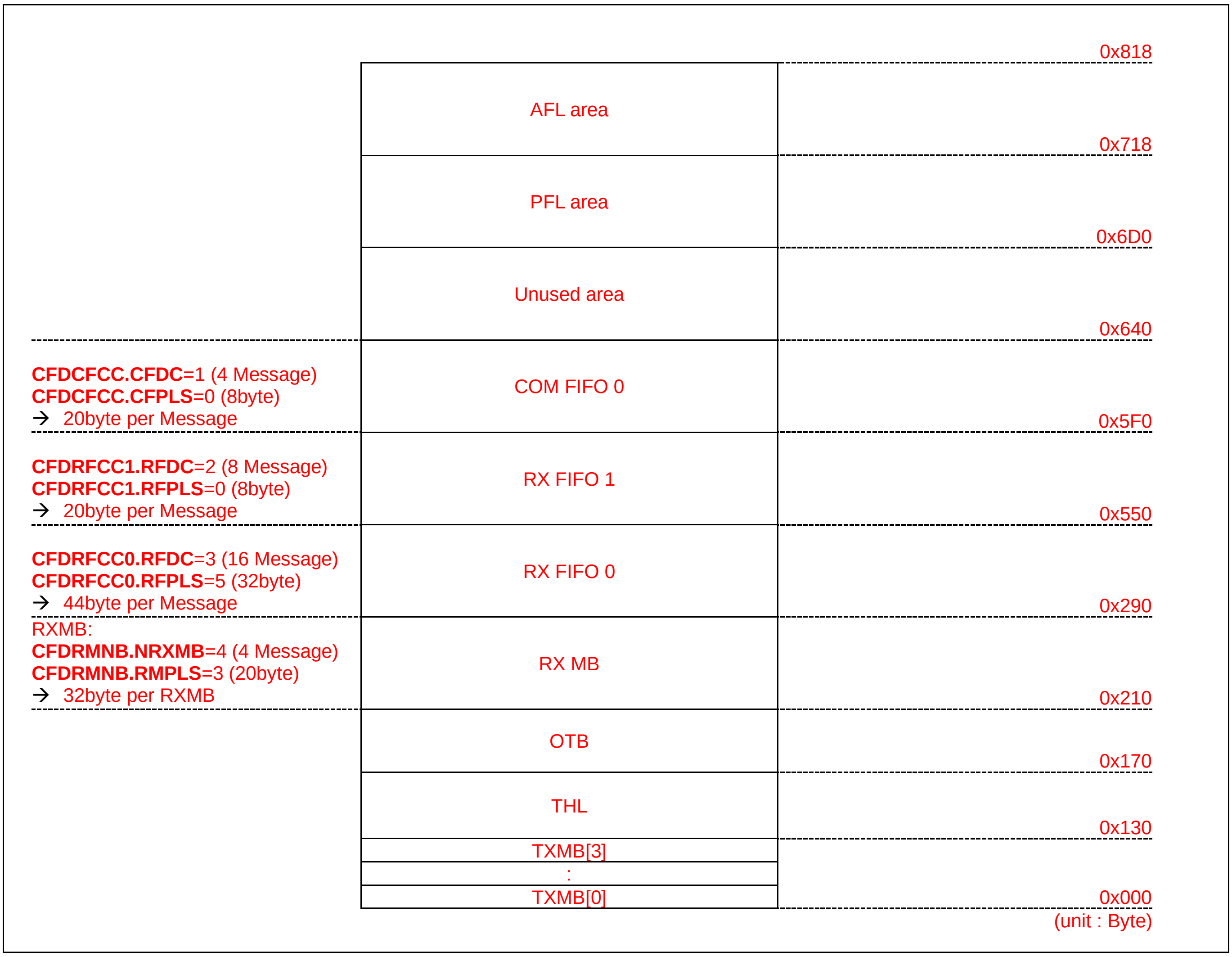


Figure 34.55 RX MB + FIFO buffers RAM area configuration examples

Added 34.10.2 OTB Area

Before correction

(no description)

After correction

The OTB area starts immediately after the area allocated for THL Buffers. The OTB is a special purpose buffer used by the CANFD. This section of RAM area can be accessed only by the CPU in RAM Test mode. Buffer needs 80 Bytes and the average number of buffers is 2. Hence, the total number of Bytes allocated for the OTB is 2*80 Bytes.

Added 34.10.3 RAM initialization cycle

Before correction

(no description)

After correction

The number of RAM initialization cycles and the RAM number of pages are shown below.

MRAM area size	RAM initialization cycles	RAM Test RTMPS range
2072	520	0x0 .. 0x8

(PCLKA cycle)

34.10 Usage notes

[Page 1903]

Before correction

34.10 Usage notes

34.10.1 Module-stop function

CANFD operation can be disabled or enabled using Module Stop Control Register C (MSTPCRC). The CANFD module is initially stopped after reset. Releasing the module-stop state enables access to the registers. For details, see section 10, Low Power Modes.

After correction

34.11 Usage notes

34.11.1 Module-stop function

CANFD operation can be disabled or enabled using Module Stop Control Register C (MSTPCRC). The CANFD module is initially stopped after reset. Releasing the module-stop state enables access to the registers. For details, see section 10, Low Power Modes.

37. Octal Serial Peripheral Interface (OSPI)

37.1 Overview

[Page 2022]

Before correction

The xSPI (eXpanded Serial Peripheral Interface) protocol specifies the interface for Non-Volatile Memory Devices, which provides high data throughput, low signal count, and limited backward compatibility with legacy SPI devices. The electrical interface can deliver up to 200 Mbytes per second raw data throughput. The OSPI is compliant with JEDEC standard JESD251(Profile 1.0 and 2.0), JESD251-1 and JESD252.

JESD251 specifies two interface profiles where profile 1.0 is Octal SPI and profile 2.0 is HyperBus™

After correction

The xSPI (eXpanded Serial Peripheral Interface) protocol specifies the interface for Non-Volatile Memory Devices, which provides high data throughput, low signal count, and limited backward compatibility with legacy SPI devices. The electrical interface can deliver up to 200 Mbytes per second raw data throughput. The OSPI is compliant with JEDEC standard JESD251(Profile 1.0 and 2.0), JESD251-1 and JESD252.

JESD251 specifies two interface profiles where profile 1.0 is Octal SPI and profile 2.0 is HyperBus™ (HyperRAM™ and HyperFlash™).

OSPI supports QSPI protocol.

Table 37.11 Data write possibility for each bus master

[Page 2079]

Before correction

Bus master	Combination Enable	Combination Disable
CPU under 32 bit Access	Not possible	Possible
CPU 64 bit Access	Possible	Possible
DMAC/DTC	Possible	Possible
EDMAC	Possible	Possible
CEU	Possible	Possible
DRW	Possible	Possible

After correction

Bus master	Combination Enable	Combination Disable
CPU under 64 bit Access	Not possible	Possible
CPU 64 bit Access	Possible	Possible
DMAC/DTC	Possible	Possible
EDMAC	Possible	Possible
CEU	Possible	Possible
DRW	Possible	Possible

38. Decryption On The Fly (DOTF)

Table 38.1 DOTF Specification

[Page 2080]

Before correction

Item	Description
Clock Source	Register clock : PCLKB AES core clock : PCLKA
AES core function	● Utilize for on-the-fly decryption of encrypted software stored in external memory. ● Block size: 128-bit ● Key size: 128-bit, 192-bit, 256-bit ● Support the following block cipher mode. − Counter (CTR) mode following NIST SP800-38A ● Support side channel counter measure function. ● Supports self-test function.
Tamper Resistance	Countermeasures available for side-channel attacks, including SPA/DPA and timing attacks
Module-stop function	Module-stop state can be set to reduce power consumption. same as OSPI module stop
Trust Zone Filter	Security and Privilege attribution can be set for each channel. same as TZF of OSPI

After correction

Item	Description
Clock Source	Register clock : PCLKB AES core clock : PCLKA
AES core function	● Utilize for on-the-fly decryption of encrypted software stored in external memory. ● Block size: 128-bit ● Key size: 128-bit, 192-bit, 256-bit ● Support the following block cipher mode. − Counter (CTR) mode following NIST SP800-38A ● Support side channel counter measure function. ● Supports self-test function. ● Counter[127:0] = {IV[127:28], Address[31:4]}, where Address is the memory mapped address of the encrypted data, aligned to the AES block size of 128 bits (16 bytes). The IV shall be chosen according to the recommendations provided in appendix B of

	SP800-38A. See section 37, Octal Serial Peripheral Interface (OSPI) and Section 4.1. Address Space for details. See NIST SP800-38A for details of AES counter mode operation.
Tamper Resistance	Countermeasures available for side-channel attacks, including SPA/DPA and timing attacks
Module-stop function	Module-stop state can be set to reduce power consumption. same as OSPI module stop
Trust Zone Filter	Security and Privilege attribution can be set for each channel. same as TZF of OSPI

39. Serial Sound Interface Enhanced (SSIE)

39.2.1 SSICR : Control Register

[Page 2089]

Before correction

23:22	FRM[1:0]	Selects Frame Word Number*1	R/W

	Communication format (SSIOFR.OMOD[1:0])		
FRM[1:0]	I²S (00b)	Monaural (10b)	TDM (01b)
00b	2	1	Setting prohibited
01b	Setting prohibited	Setting prohibited	4
10b			5
11b			6

After correction

23:22	FRM[1:0]	Selects Frame Word Number*1	R/W

	Communication format (SSIOFR.OMOD[1:0])		
FRM[1:0]	I²S (00b)	Monaural (10b)	TDM (01b)
00b	2	1	Setting prohibited
01b	Setting prohibited	Setting prohibited	4
10b			6
11b			8

Table 39.7 Bits subject to software reset by the RFRST bit

[Page 2110]

Before correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	0x00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	0x04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST

SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	0x1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

After correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	0x00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	0x04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	SSIFTDR[31:16]															
		+2	SSIFTDR[15:0]															
SSIFRDR	0x1C	+0	SSIFRDR[31:16]															
		+2	SSIFRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

Table 39.8 Bits subject to software reset by the TFRST bit (2 of 2)

. [Page 2111]

Before correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST

		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	0x1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

After correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	SSIFTDR[31:16]															
		+2	SSIFTDR[15:0]															
SSIFRDR	0x1C	+0	SSIFRDR[31:16]															
		+2	SSIFRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

Table 39.9 Bits subject to software reset by the SSIRST bit

[Page 2113]

Before correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	0x00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	0x04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	0x1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

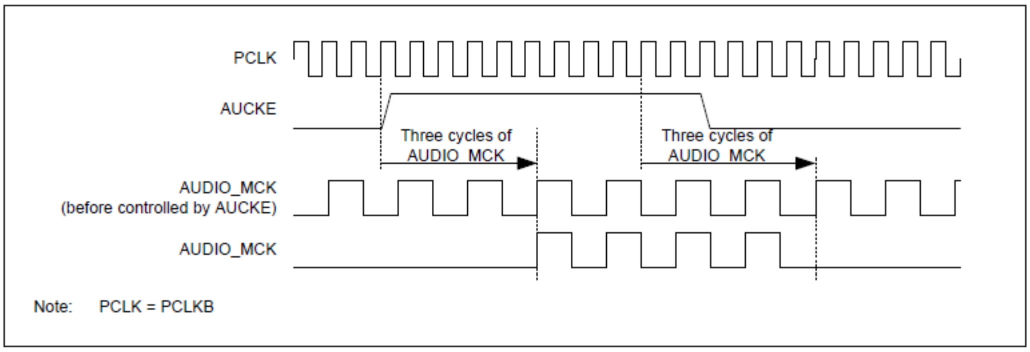
After correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	0x00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	0x04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	SSIFTDR[31:16]															
		+2	SSIFTDR[15:0]															
SSIFRDR	0x1C	+0	SSIFRDR[31:16]															
		+2	SSIFRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

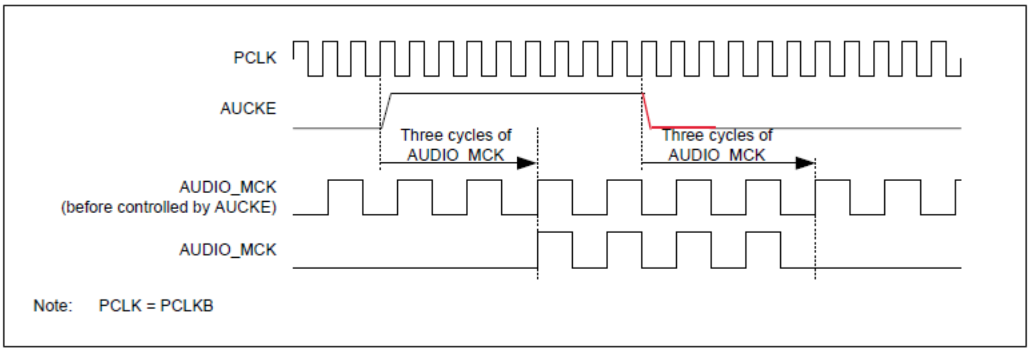
Figure 39.25 Stop/resume of AUDIO_MCK

[Page 2114]

Before correction



After correction



39.2.4 SSIFSR : FIFO Status Register

[Page 2115 to 2118]

Before correction

Bit	Symbol	Function	R/W
0	RDF	Receive Data Full Flag 0: The size of received data in SSIFRDR is not more than the value of SSISCR.RDFS. 1: The size of received data in SSIFRDR is not less than the value of SSISCR.RDFS plus one.	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W
13:8	RDC[5:0]	Number of Receive FIFO Data Indication Flag Number of receive FIFO data indication flag	R
15:14	—	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Function	R/W
16	TDE	Transmit Data Empty Flag 0: The free space of SSIFTDR is not more than the value of SSISCR.TDES. 1: The free space of SSIFTDR is not less than the value of SSISCR.TDES plus one.	R/W
23:17	—	These bits are read as 0. The write value should be 0.	R/W
29:24	TDC[5:0]	Number of Transmit FIFO Data Indication Flag Number of transmit FIFO data indication flag	R
31:30	—	These bits are read as 0. The write value should be 0.	R/W

RDF flag (Receive Data Full Flag)

The RDF flag indicates that the receive FIFO data register (SSIFRDR) has unread received data not less than the amount set with the SSISCR.RDFS bit plus one. This flag is set by automatic determination but it must be cleared by register access.

(omission)

[Setting condition]

SSIFTDR has free space not less than the amount set with the SSIFCR.TTRG bit plus one.

[Setting timing]

At completion of transfer from the shift register that results in SSIFRDR having data not less than the amount set with the SSISCR.RDFS bit plus one.

RDC[5:0] flags (Number of Receive FIFO Data Indication Flag)

The RDC[5:0] flags indicate the number of valid data that are stored in the receive FIFO data register (SSIFRDR). With this flag as 0x00, there is no received data. With 0x20, the register is filled with received data and there is no free space.

TDE flag (Transmit Data Empty Flag)

The TDE flag indicates that the transmit FIFO data register (SSIFTDR) has free space not less than the amount set with the SSIFCR.TTRG bit plus one. This flag is set by automatic determination but it must be cleared by register access.

(omission)

[Setting condition]

SSIFTDR has free space not less than the amount set with the SSIFCR.TTRG bit plus one.

[Setting timing]

While operating on PCLKB, SSIFTDR is found to have free space not less than “size set in the SSISCR.TDES bits + 1.”

TDC[5:0] flags (Number of Transmit FIFO Data Indication Flag)

The TDC[5:0] flags indicate the number of valid data that are stored in the transmit FIFO data register (SSIFTDR). With this flag as 0x00, there is no data to be transmitted. With 0x20, there is no space to write data.

After correction

Bit	Symbol	Function	R/W
0	RDF	Receive Data Full Flag 0: The size of received data in SSIFRDR is not more than the value of SSISCR.RDFS. 1: The size of received data in SSIFRDR is not less than the value of SSISCR.RDFS plus one.	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W
13:8	RDC[5:0]	Receive Data Count Number of valid data stored in the receive FIFO data register	R
15:14	—	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Function	R/W
16	TDE	Transmit Data Empty Flag 0: The free space of SSIFTDR is not more than the value of SSISCR.TDES. 1: The free space of SSIFTDR is not less than the value of SSISCR.TDES plus one.	R/W
23:17	—	These bits are read as 0. The write value should be 0.	R/W
29:24	TDC[5:0]	Transmit Data Count Number of valid data stored in the transmit FIFO data register	R
31:30	—	These bits are read as 0. The write value should be 0.	R/W

RDF flag (Receive Data Full Flag)

The RDF flag indicates that the receive FIFO data register (SSIFRDR) has unread received data not less than the amount set with the SSISCR.RDFS bits plus one. This flag is set by automatic determination but it must be cleared by register access.

(omission)

[Setting condition]

SSIFRDR has free space not less than the amount set with the SSISCR.RDFS bits plus one.

[Setting timing]

At completion of transfer from the shift register that results in SSIFRDR having data not less than the amount set with the SSISCR.RDFS bits plus one.

RDC[5:0] bits (Receive Data Count)

The RDC[5:0] bits indicate the number of valid data that are stored in the receive FIFO data register (SSIFRDR). With these bits as 0x00, there is no received data. With 0x20, the register is filled with received data and there is no free space.

TDE flag (Transmit Data Empty Flag)

The TDE flag indicates that the transmit FIFO data register (SSIFTDR) has free space not less than the amount set with the SSISCR.TDES bits plus one. This flag is set by automatic determination but it must be cleared by register access.

(omission)

[Setting condition]

SSIFTDR has free space not less than the amount set with the SSISCR.TDES bits plus one.

[Setting timing]

While operating on PCLKB, SSIFTDR is found to have free space not less than “size set in the SSISCR.TDES bits + 1.”

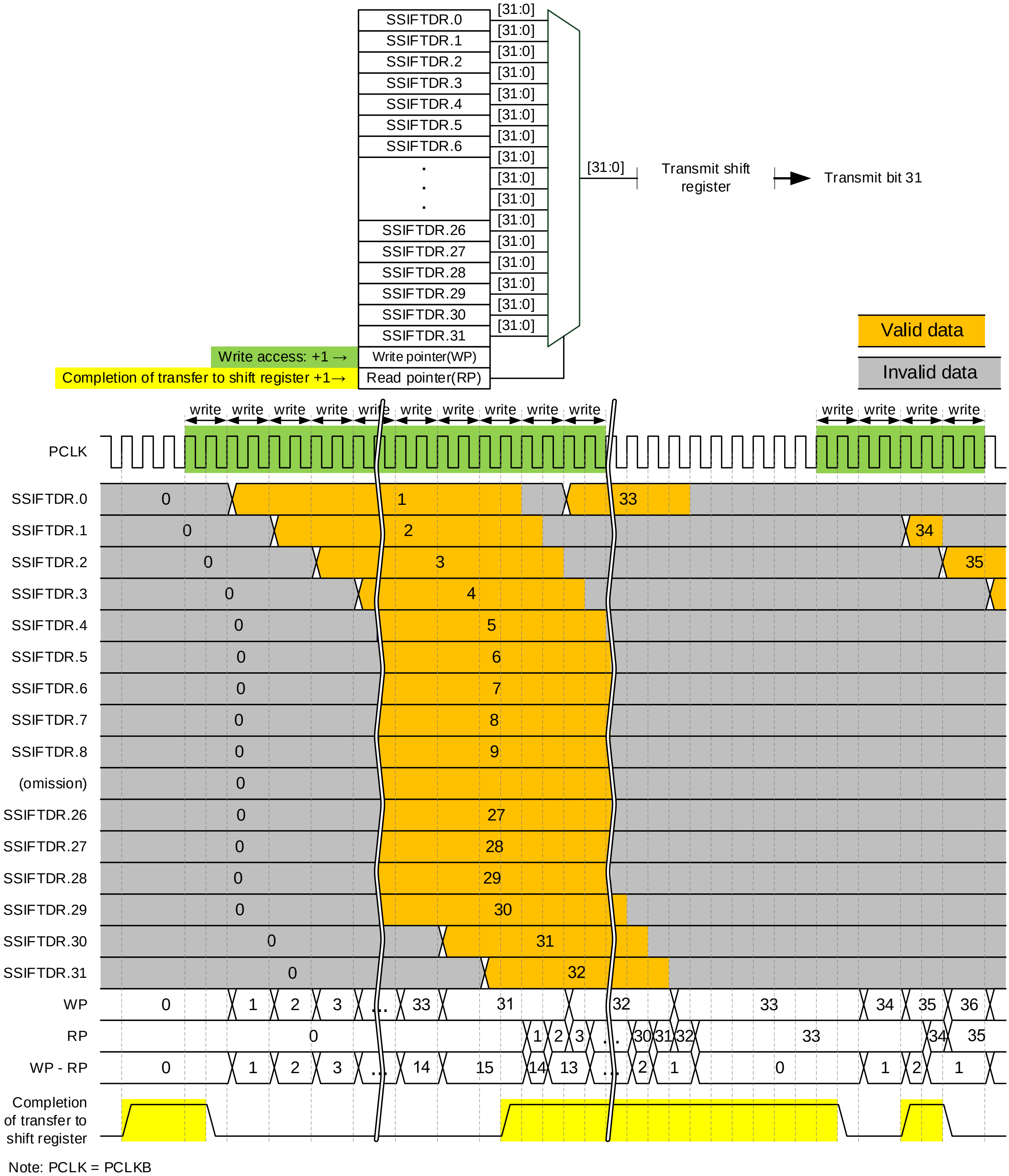
TDC[5:0] bits (Transmit Data Count)

The TDC[5:0] bits indicate the number of valid data that are stored in the transmit FIFO data register (SSIFTDR). With these bits as 0x00, there is no data to be transmitted. With 0x20, there is no space to write data.

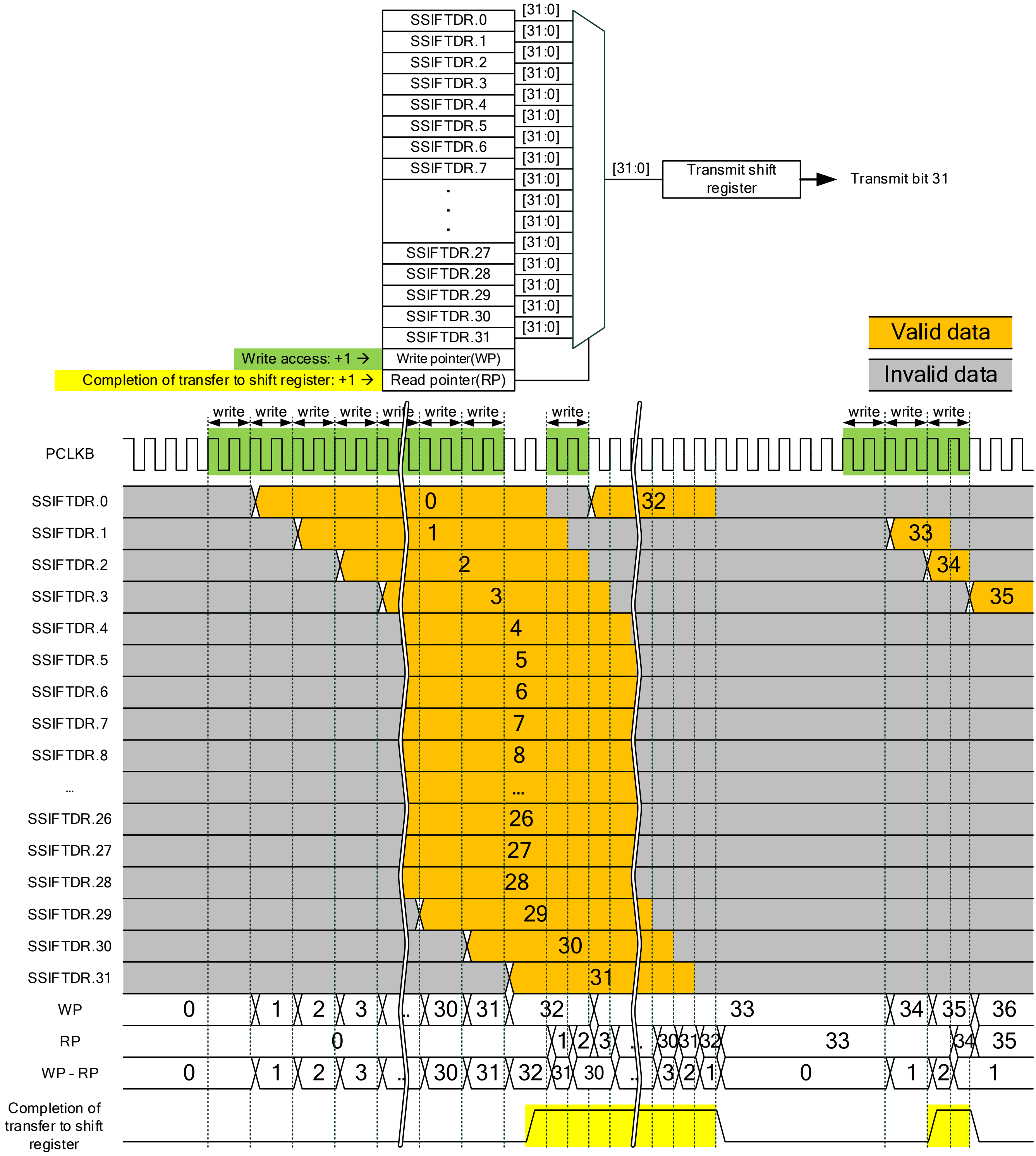
Figure 39.31 Configuration of the transmit FIFO data register and transmit shift register, and FIFO operation example

[Page 2120]

Before correction



After correction



Note: PCLK = PCLKB

39.2.6 SSIFRDR : Receive FIFO Data Register

[Page 2121]

Before correction

Figure 39.31 shows the configurations and operation examples of the receive FIFO data register and receive shift register.

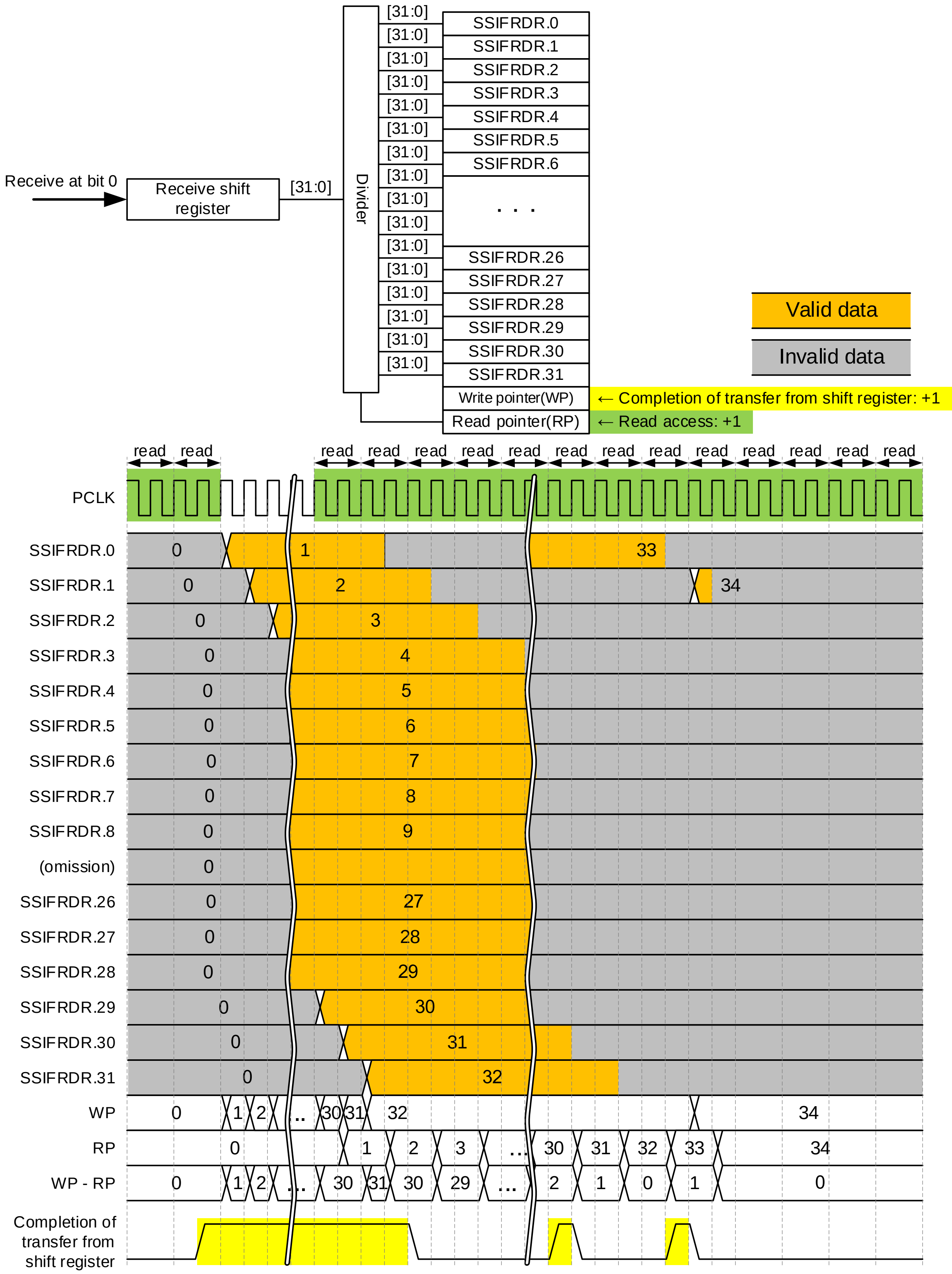
After correction

Figure 39.32 shows the configurations and operation examples of the receive FIFO data register and receive shift register.

Figure 39.32 Configuration of the transmit FIFO data register and transmit shift register, and FIFO operation example

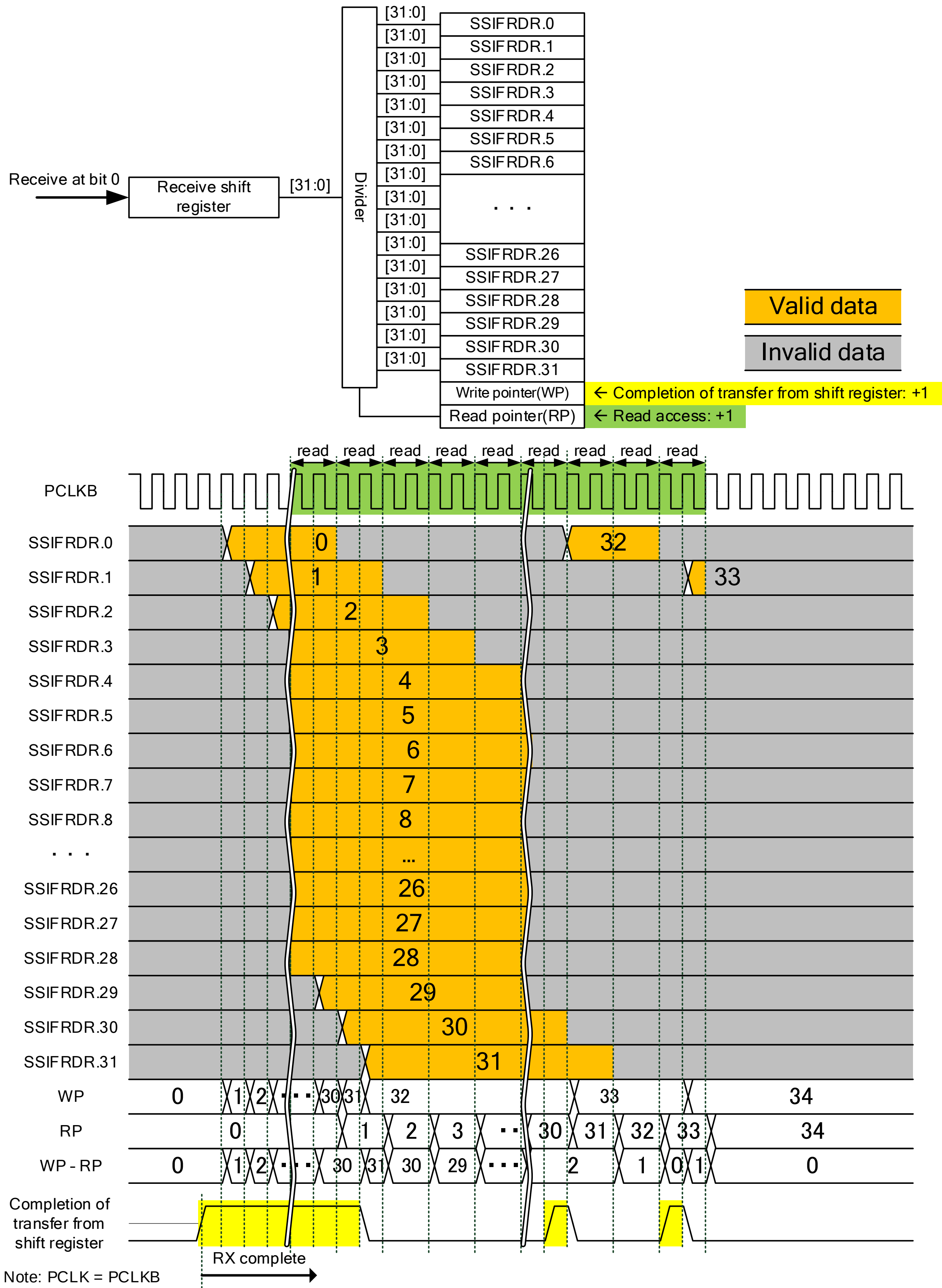
[Page 2122]

Before correction



Note: PCLK = PCLKB

After correction



Title: Configuration of the receive FIFO data register and receive shift register, and FIFO operation example

39.5.2.1 Data communication state

[Page 2136]

Before correction

● State Transition in the Setting with Padding Bits

When SSIE ends transfer of the last bit of a data word during communication (SSISR.IIRQ = 0), SSIE transitions from the data communication state to the padding communication state in Figure 39.49. Except in the status with SSICR.SDTA = 1 and transmission and reception disabled (SSICR.TEN = 0 and SSICR.REN = 0), SSIE transitions from the data communication state to the idle state when it stops communication in Figure 39.51.

After correction

● State Transition in the Setting with Padding Bits

When SSIE ends transfer of the last bit of a data word during communication (SSISR.IIRQ = 0), SSIE transitions from the data communication state to the padding communication state in Figure 39.49. Except in the status with SSICR.SDTA = 1 and transmission and reception disabled (SSICR.TEN = 0 and SSICR.REN = 0), SSIE transitions from the data communication state to the idle state when it stops communication in Figure 39.50.

39.7.1 SSIEn_SSIF Interrupt (n = 0, 1)

[Page 2145]

Before correction

● Transmit underflow interrupt

As the transmit underflow interrupt, SSISR.TUIRQ is output while SSICR.TUIEN = 1. When you use SSIE for transmission, enable the output of this interrupt (SSICR.TUIRQ = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

● Transmit overflow interrupt

As the transmit overflow interrupt, SSISR.TOIRQ is output while SSICR.TOIRQ = 1. When you use SSIE for transmission, enable the output of this interrupt (SSICR.TOIRQ = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

● Receive underflow interrupt

As the receive underflow interrupt, SSISR.RUIRQ is output while SSICR.RUIRQ = 1. When you use SSIE for reception, enable the output of this interrupt (SSICR.RUIRQ = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

● Receive overflow interrupt

As the receive overflow interrupt, SSISR.ROIRQ is output while SSICR.ROIRQ = 1. When you use SSIE for reception, enable the output of this interrupt (SSICR.ROIRQ = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

After correction

● Transmit underflow interrupt

As the transmit underflow interrupt, SSISR.TUIRQ is output while SSICR.TUIEN = 1. When you use SSIE for transmission, enable the output of this interrupt (SSICR.TUIEN = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

- Transmit overflow interrupt

As the transmit overflow interrupt, SSISR.TOIRQ is output while SSICR.TOIEN = 1. When you use SSIE for transmission, enable the output of this interrupt (SSICR. TOIEN = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

- Receive underflow interrupt

As the receive underflow interrupt, SSISR.RUIRQ is output while SSICR.RUIEN = 1. When you use SSIE for reception, enable the output of this interrupt (SSICR. RUIEN = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

- Receive overflow interrupt

As the receive overflow interrupt, SSISR.ROIRQ is output while SSICR.ROIEN = 1. When you use SSIE for reception, enable the output of this interrupt (SSICR. ROIEN = 1). If this interrupt occurs, follow instructions in the procedure to halt communication in Figure 39.56 and error-handling procedure in Figure 39.57.

Table 39.19 Bits protected from writing during communication

[Page 2151]

Before correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	0x00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	0x04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	0x1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

RENESAS

After correction

Symbol	Address (BASE+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	0x00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	0x04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	0x10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	—	—	—	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	0x14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	0x18	+0	SSIFTDR[31:16]															
		+2	SSIFTDR[15:0]															
SSIFRDR	0x1C	+0	SSIFRDR[31:16]															
		+2	SSIFRDR[15:0]															
SSIOFR	0x20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	0x24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

43. Security Features

43.3.3.5 Memory Security Attribution of TrustZone filter

[Page 2232]

Before correction

The contents in Secure or Non-secure Callable regions must be the same in both bank0 and bank1 in the dual mode. Otherwise, the contents of secure or non-secure regions may not be consistent after a field update.

After correction

Bank swapping by the non-secure user may change the code in the secure or non-secure callable areas. The following is recommended to prevent this.

・Set BANKSEL_SEL.BANKSWP[2:0]=000b to disable the bank swapping for non-secure users.

・Make the code in secure or non-secure callable areas for both banks the same when allow non-secure user to swap banks.

45. 12-Bit A/D Converter (ADC12)

45.6.13 Port Settings When Using the ADC12 Input

[Page 2367]

Before correction

When using the high-precision channels, do not use PORT0 as general I/O.

After correction

When using the high-precision channels, do not use PORT0 as digital output ports.

52. Flash Memory

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

[Page 2437]

Before correction

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 . The first character ("R",0x52 in ASCII code) of the part number is stored in the byte with the smallest address (0x0300_80F0). When reading by the signature request command of the serial programming interface, the data is read in order from the data with the small address. That is, the data in 0x0300_80F0 is read first, and in 0x0300_80FF is read last.

After correction

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 Table 1.13 Product list. The first character ("R",0x52 in ASCII code) of the part number is stored in the byte with the smallest address (0x0300_80F0). When reading by the signature request command of the serial programming interface, the data is read in order from the data with the small address. That is, the data in 0x0300_80F0 is read first, and in 0x0300_80FF is read last.

60. Electrical Characteristics

Table 60.8 Current of high-speed mode, maximum condition (DCDC mode)

[Page 2842]

Before correction

Parameter		Symbol	Typ	Max		Unit	Test conditions
				105℃	125℃		
Supply current *1*2	—	I_{CC}	—	5.97	6.11	mA	

After correction

Parameter		Symbol	Typ	Max		Unit	Test conditions
				105℃	125℃		
Supply current *1*2	—	I_{CC}	—	7.05	7.19	mA	

Table 60.9 Current of high-speed mode, maximum condition (External VDD mode)

[Page 2843]

Before correction

Parameter	CPUCLK Frequency	Symbol	Typ	Max		Unit	Test conditions
				105℃	125℃		
Supply current*1 *2	—	I_{CC}	—	5.97	6.11	mA	

After correction

Parameter	CPUCLK Frequency	Symbol	Typ	Max		Unit	Test conditions
				105℃	125℃		
Supply current*1 *2	—	I_{CC}	—	7.05	7.19	mA	