Date: May 13, 2025

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*-A0147A/E	Rev.	1.00
Title	Notes on switching the peripheral clock division ratio from other than 1		Information Category	Technical Notification		
Applicable Product	RA6M4, RA6M5, RA4M2, RA4M3, RA6E1, RA4E1, RA6T2, RA6T3, RA6E2, RA4E2, RA8M1, RA8D1, RA8E1, RA8T1, RA8E2	Lot No. All	Reference Document	Refer table at the end of this document		

User's Manual will be revised as follows

1) RA6M4
2) RA6M5
3) RA4M2, RA4M3, RA6E1, RA4E1
4) RA6T2
5) RA6T3
6) RA6E2, RA4E2
7) RA8M1
8) RA8D1
9) RA8E1
10) RA8T1
11) RA8E2

8. Clock Generation Circuit

1) RA6M4
2) RA6M5
3) RA4M2, RA4M3, RA6E1, RA4E1
5) RA6T3
6) RA6E2, RA4E2
7) RA8M1
8) RA8D1
9) RA8E1
10) RA8T1
11) RA8E2

8.2.28 USBCKDIVCR : USB Clock Division Control Register for RA6M4, RA6M5
8.2.26 USBCKDIVCR : USB Clock Division Control Register for RA4M2, RA4M3, RA6E1, RA4E1
8.2.24 USBCKDIVCR : USB Clock Division Control Register for RA6T3, RA6E2, RA4E2
8.2.36 USBCKDIVCR : USB Clock Division Control Register for RA8M1, RA8T1, RA8D1, RA8E2
8.2.33 USBCKDIVCR : USB Clock Division Control Register for RA8E1

Before Correction

USBCKDIV[2:0] bits (USB Clock (USBCLK) Division Select)

These bits select the frequency of the USB clock (USBCLK) and must be modified when USBCKCR.USBCKSRDY = 1.

After Correction

USBCKDIV[2:0] bits (USB Clock (USBCLK) Division Select)

These bits select the frequency of the USB clock (USBCLK) and must be modified when USBCKCR.USBCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set all of MSTPCRB.MSTPBi (i=11,12) = 1 before setting USBCKCR.USBCKSREQ = 1.

8.2.30 USBCKCR : USB Clock Control Register for RA6M4
8.2.33 USBCKCR : USB Clock Control Register for RA6M5
8.2.27 USBCKCR : USB Clock Control Register for RA4M2, RA4M3, RA6E1, RA4E1
8.2.27 USBCKCR : USB Clock Control Register for RA6T3
8.2.28 USBCKCR : USB Clock Control Register for RA6E2, RA4E2
8.2.41 USBCKCR : USB Clock Control Register for RA8M1, RA8D1
8.2.36 USBCKCR : USB Clock Control Register for RA8E1
8.2.39 USBCKCR : USB Clock Control Register for RA8T1, RA8E2

Before Correction

The USBCKCR register controls the USB clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate

stable output. To change the set value of USBCKDIVCR.USBCKDIV[2:0] and USBCKSEL[2:0], use the following procedure:

1. Write 1 to USBCKSREQ.
2. Poll until USBCKSRDY is read as 1. While USBCKSRDY = 1, no clock is output to USBCLK.
3. Write to USBCKDIVCR.USBCKDIV[2:0] and USBCKSEL[2:0].
4. Write 0 to USBCKSREQ.
5. Poll until USBCKSRDY is read as 0.
6. When USBCKSRDY becomes 0, USBCLK starts to output. Clock switching is complete.

After Correction

The USBCKCR register controls the USB clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of USBCKDIVCR.USBCKDIV[2:0] and USBCKSEL[2:0], use the following procedure:

1. Write 1 to all of MSTPCRB.MSTPBi (i=11,12) (only when switching the division ratio setting from n (n≠1) to m (m≠1))
2. Wait for two USBCLK (only when switching the division ratio setting from n (n≠1) to m (m≠1))
3. Write 1 to USBCKSREQ.
4. Poll until USBCKSRDY is read as 1. While USBCKSRDY = 1, no clock is output to USBCLK.
5. Write to USBCKDIVCR.USBCKDIV[2:0] and USBCKSEL[2:0].
6. Write 0 to USBCKSREQ.
7. Poll until USBCKSRDY is read as 0.
8. When USBCKSRDY becomes 0, USBCLK starts to output. Clock switching is complete.

1) RA6M4
2) RA6M5

8.2.29 OCTACKDIVCR : Octal-SPI Clock Division Control Register for RA6M4, RA6M5

Before Correction

OCTACKDIV[2:0] bits (Octal-SPI Clock (OCTACLK) Division Select)

These bits select the frequency of the Octal-SPI clock (OCTACLK) and must be modified when OCTACKCR.OCTACKSRDY = 1.

After Correction

OCTACKDIV[2:0] bits (Octal-SPI Clock (OCTACLK) Division Select)

These bits select the frequency of the Octal-SPI clock (OCTACLK) and must be modified when OCTACKCR.OCTACKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRB.MSTPB16 = 1 before setting OCTACKCR.OCTACKSREQ = 1.

8.2.31 OCTACKCR : Octal-SPI Clock Control Register for RA6M4

8.2.34 OCTACKCR : Octal-SPI Clock Control Register for RA6M5

Before Correction

The OCTACKCR register controls the Octal-SPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0], use the following procedure:

1. Write 1 to OCTACKSREQ.
2. Poll until OCTACKSRDY is read as 1. While OCTACKSRDY = 1, no clock is output to OCTACLK.
3. Write to OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0].
4. Write 0 to OCTACKSREQ.
5. Poll until OCTACKSRDY is read as 0.
6. When OCTACKSRDY becomes 0, OCTACLK starts to output. Clock switching is complete.

After Correction

The OCTACKCR register controls the Octal-SPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRB.MSTPB16 (only when switching the division ratio setting from n (n≠1) to m (m≠1))
2. Wait for two OCTACLK (only when switching the division ratio setting from n (n≠1) to m (m≠1))
3. Write 1 to OCTACKSREQ.
4. Poll until OCTACKSRDY is read as 1. While OCTACKSRDY = 1, no clock is output to OCTACLK.
5. Write to OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0].
6. Write 0 to OCTACKSREQ.
7. Poll until OCTACKSRDY is read as 0.
8. When OCTACKSRDY becomes 0, OCTACLK starts to output. Clock switching is complete.

7) RA8M1
8) RA8D1
9) RA8E1

8.2.37 OCTACKDIVCR : Octal-SPI Clock Division Control Register for RA8M1, RA8D1
8.2.34 OCTACKDIVCR : Octal-SPI Clock Division Control Register for RA8E1

Before Correction

OCTACKDIV[2:0] bits (Octal-SPI Clock (OCTACLK) Division Select)

These bits select the frequency of the Octal-SPI clock (OCTACLK) and must be modified when OCTACKCR.OCTACKSRDY = 1.

After Correction

OCTACKDIV[2:0] bits (Octal-SPI Clock (OCTACLK) Division Select)

These bits select the frequency of the Octal-SPI clock (OCTACLK) and must be modified when OCTACKCR.OCTACKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRB.MSTPB16 = 1

before setting OCTACKCR.OCTACKSREQ = 1.

8.2.42 OCTACKCR : Octal-SPI Clock Control Register for RA8M1, RA8D1

8.2.37 OCTACKCR : Octal-SPI Clock Control Register for RA8E1

Before Correction

The OCTACKCR register controls the Octal-SPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0], use the following procedure:

1. Write 1 to OCTACKSREQ.
2. Poll until OCTACKSRDY is read as 1. While OCTACKSRDY = 1, no clock is output to OCTACLK.
3. Write to OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0].
4. Write 0 to OCTACKSREQ.
5. Poll until OCTACKSRDY is read as 0.
6. When OCTACKSRDY becomes 0, OCTACLK starts to output. Clock switching is complete.

After Correction

The OCTACKCR register controls the Octal-SPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRB.MSTPB16 (only when switching the division ratio setting from n (n≠1))
2. Wait for two OCTACLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to OCTACKSREQ.
4. Poll until OCTACKSRDY is read as 1. While OCTACKSRDY = 1, no clock is output to OCTACLK.
5. Write to OCTACKDIVCR.OCTACKDIV[2:0] and OCTACKSEL[2:0].
6. Write 0 to OCTACKSREQ.
7. Poll until OCTACKSRDY is read as 0.
8. When OCTACKSRDY becomes 0, OCTACLK starts to output. Clock switching is complete.

2) RA6M5
4) RA6T2
5) RA6T3
6) RA6E2, RA4E2
7) RA8M1
8) RA8D1
9) RA8E1
10) RA8T1
11) RA8E2

8.2.30 CANFDCKDIVCR : CANFD Clock Division Control Register for RA6M5

8.2.25 CANFDCKDIVCR : CANFD Clock Division Control Register for RA6T3

8.2.23 CANFDCKDIVCR : CANFD Clock Division Control Register for RA6T2
8.2.25 CANFDCKDIVCR : CANFD Clock Division Control Register for RA6E2, RA4E2
8.2.38 CANFDCKDIVCR : CANFD Clock Division Control Register for RA8M1, RA8D1, RA8E2
8.2.35 CANFDCKDIVCR : CANFD Clock Division Control Register for RA8E1
8.2.37 CANFDCKDIVCR : CANFD Clock Division Control Register for RA8T1

Before Correction

CANFDCKDIV[2:0] bits (CANFD Clock (CANFDCLK) Division Select)

These bits select the frequency of the CANFD clock (CANFDCLK) and must be modified when CANFDCKCR.CANFDCKSRDY = 1.

After Correction

CANFDCKDIV[2:0] bits (CANFD Clock (CANFDCLK) Division Select)

These bits select the frequency of the CANFD clock (CANFDCLK) and must be modified when CANFDCKCR.CANFDCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRC.MSTPC27 = 1 before setting CANFDCKCR.CANFDCKSREQ = 1.

8.2.35 CANFDCKCR : CANFD Clock Control Register for RA6M5
8.2.28 CANFDCKCR : CANFD Clock Control Register for RA6T3
8.2.27 CANFDCKCR : CANFD Clock Control Register for RA6T2
8.2.29 CANFDCKCR : CANFD Clock Control Register for RA6E2, RA4E2
8.2.43 CANFDCKCR : CANFD Clock Control Register for RA8M1, RA8D1
8.2.38 CANFDCKCR : CANFD Clock Control Register for RA8E1
8.2.40 CANFDCKCR : CANFD Clock Control Register for RA8T1
8.2.41 CANFDCKCR : CANFD Clock Control Register for RA8E2

Before Correction

The CANFDCKCR register controls the CANFD clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of CANFDCKDIVCR.CANFDCKDIV[2:0] and CANFDCKSEL[2:0], use the following procedure:

1. Write 1 to CANFDCKSREQ.
2. Poll until CANFDCKSRDY is read as 1. While CANFDCKSRDY = 1, no clock is output to CANFDCLK.
3. Write to CANFDCKDIVCR. CANFDCKDIV[2:0] and CANFDCKSEL[2:0].
4. Write 0 to CANFDCKSREQ.
5. Poll until CANFDCKSRDY is read as 0.
6. When CANFDCKSRDY becomes 0, CANFDCLK starts to output. Clock switching is complete.

After Correction

The CANFDCKCR register controls the CANFD clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of CANFDCKDIVCR. CANFDCKDIV[2:0] and CANFDCKSEL[2:0], use the

following procedure:

1. Write 1 to MSTPCRC.MSTPC27 (only when switching the division ratio setting from n (n≠1))
2. Wait for two CANFDCLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to CANFDCKSREQ.
4. Poll until CANFDCKSRDY is read as 1. While CANFDCKSRDY = 1, no clock is output to CANFDCLK.
5. Write to CANFDCKDIVCR. CANFDCKDIV[2:0] and CANFDCKSEL[2:0].
6. Write 0 to CANFDCKSREQ.
7. Poll until CANFDCKSRDY is read as 0.
8. When CANFDCKSRDY becomes 0, CANFDCLK starts to output. Clock switching is complete.

2) RA6M5

7) RA8M1

8) RA8D1

8.2.31 USB60CKDIVCR : USB60 Clock Division Control Register for RA6M5

8.2.36 USB60CKDIVCR : USB60 Clock Division Control Register for RA8M1、RA8D1

Before Correction

USB60CKDIV[2:0] bits (USB60 Clock (USB60CLK) Division Select)

These bits select the frequency of the USB60 clock (USB60CLK) and must be modified when USB60CKCR.USB60CKSRDY = 1.

After Correction

USB60CKDIV[2:0] bits (USB60 Clock (USB60CLK) Division Select)

These bits select the frequency of the USB60 clock (USB60CLK) and must be modified when USB60CKCR.USB60CKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRB.MSTPB12 = 1 before setting USB60CKCR.USB60CKSREQ = 1.

8.2.39 USB60CKCR : USB60 Clock Control Register for RA6M5

8.2.44 USB60CKCR : USB60 Clock Control Register for RA8M1, RA8D1

Before Correction

The USB60CKCR register controls the USB60 clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of USB60CKDIVCR.USB60CKDIV[2:0] and USB60CKSEL[2:0], use the following procedure:

1. Write 1 to USB60CKSREQ.
2. Poll until USB60CKSRDY is read as 1. While USB60CKSRDY = 1, no clock is output to USB60CLK.
3. Write to USB60CKDIVCR.USB60CKDIV[2:0] and USB60CKSEL[2:0].
4. Write 0 to USB60CKSREQ.
5. Poll until USB60CKSRDY is read as 0.
6. When USB60CKSRDY becomes 0, USB60CLK starts to output. Clock switching is complete.

After Correction

The USB60CKCR register controls the USB60 clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of USB60CKDIVCR.USB60CKDIV[2:0] and USB60CKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRB.MSTPB12 (only when switching the division ratio setting from n (n≠1))
2. Wait for two USB60CLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to USB60CKSREQ.
4. Poll until USB60CKSRDY is read as 1. While USB60CKSRDY = 1, no clock is output to USB60CLK.
5. Write to USB60CKDIVCR.USB60CKDIV[2:0] and USB60CKSEL[2:0].
6. Write 0 to USB60CKSREQ.
7. Poll until USB60CKSRDY is read as 0.
8. When USB60CKSRDY becomes 0, USB60CLK starts to output. Clock switching is complete.

5) RA6T3
6) RA6E2, RA4E2
7) RA8M1
8) RA8D1
10) RA8T1

8.2.26 I3CCKDIVCR : I3C Clock Division Control Register for RA6T3
8.2.27 I3CCKDIVCR : I3C Clock Division Control Register for RA6E2, RA4E2
8.2.40 I3CCKDIVCR : I3C Clock Division Control Register for RA8M1, RA8D1
8.2.38 I3CCKDIVCR : I3C Clock Division Control Register for RA8T1

Before Correction

I3CCKDIV[2:0] bits (I3C Clock (I3CCLK) Division Select)

These bits select the frequency of the I3C clock (I3CCLK) and must be modified when I3CCKCR.I3CCKSRDY = 1.

After Correction

I3CCKDIV[2:0] bits (I3C Clock (I3CCLK) Division Select)

These bits select the frequency of the I3C clock (I3CCLK) and must be modified when I3CCKCR.I3CCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRB.MSTPB4 = 1 before setting I3CCKCR.I3CCKSREQ = 1.

8.2.29 I3CCKCR : I3C Clock Control Register for RA6T3
8.2.31 I3CCKCR : I3C Clock Control Register for RA6E2, RA4E2
8.2.45 I3CCKCR : I3C Clock Control Register for RA8M1, RA8D1
8.2.41 I3CCKCR : I3C Clock Control Register for RA8T1

Before Correction

The I3CCKCR register controls the I3C clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate

stable output. To change the set value of I3CCKDIVCR.I3CCKDIV[2:0] and I3CCKSEL[2:0], use the following procedure:

1. Write 1 to I3CCKSREQ.
2. Poll until I3CCKSRDY is read as 1. While I3CCKSRDY = 1, no clock is output to I3CCLK.
3. Write to I3CCKDIVCR.I3CCKDIV[2:0] and I3CCKSEL[2:0].
4. Write 0 to I3CCKSREQ.
5. Poll until I3CCKSRDY is read as 0.
6. When I3CCKSRDY becomes 0, I3CCLK starts to output. Clock switching is complete.

After Correction

The I3CCKCR register controls the I3C clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of I3CCKDIVCR.I3CCKDIV[2:0] and I3CCKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRB.MSTPB4 (only when switching the division ratio setting from n (n≠1)
2. Wait for two I3CCLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to I3CCKSREQ.
4. Poll until I3CCKSRDY is read as 1. While I3CCKSRDY = 1, no clock is output to I3CCLK.
5. Write to I3CCKDIVCR.I3CCKDIV[2:0] and I3CCKSEL[2:0].
6. Write 0 to I3CCKSREQ.
7. Poll until I3CCKSRDY is read as 0.
8. When I3CCKSRDY becomes 0, I3CCLK starts to output. Clock switching is complete.

4) RA6T2

8.2.22 SCISPICKDIVCR : SCISPI Clock Division Control Register for RA6T2

Before Correction

SCISPICKDIV[2:0] bits (SCISPI Clock (SCISPICLK) Division Select)

These bits select the frequency of the SCISPI clock (SCISPICLK) and must be modified when SCISPICKCR.SCISPICKSRDY = 1.

After Correction

SCISPICKDIV[2:0] bits (SCISPI Clock (SCISPICLK) Division Select)

These bits select the frequency of the SCISPI clock (SCISPICLK) and must be modified when SCISPICKCR.SCISPICKSRDY = 1.

When switching the division ratio setting from n (n≠1), set all of MSTPCRB.MSTPBi(i=18,19,22,27,28,29,30,31) = 1

before setting SCISPICKCR.SCISPICKSREQ = 1.

8.2.26 SCISPICKCR : SCISPI Clock Control Register for RA6T2

Before Correction

The SCISPICKCR register controls the SCISPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of SCISPICKDIVCR.SCISPICKDIV[2:0] and SCISPICKSEL[2:0], use the following procedure:

1. Write 1 to SCISPICKSREQ.
2. Poll until SCISPICKSRDY is read as 1. While SCISPICKSRDY = 1, no clock is output to SCISPICLK.
3. Write to SCISPICKDIVCR. SCISPICKDIV[2:0] and SCISPICKSEL[2:0].
4. Write 0 to SCISPICKSREQ.
5. Poll until SCISPICKSRDY is read as 0.
6. When SCISPICKSRDY becomes 0, SCISPICLK starts to output. Clock switching is complete.

After Correction

The SCISPICKCR register controls the SCISPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of SCISPICKDIVCR. SCISPICKDIV[2:0] and SCISPICKSEL[2:0], use the following procedure:

1. Write 1 to all of MSTPCRB.MSTPBi (i=18,19,22,27,28,29,30,31) (only when switching the division ratio setting from n (n≠1))
2. Wait for two SCISPICLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to SCISPICKSREQ.
4. Poll until SCISPICKSRDY is read as 1. While SCISPICKSRDY = 1, no clock is output to SCISPICLK.
5. Write to SCISPICKDIVCR. SCISPICKDIV[2:0] and SCISPICKSEL[2:0].
6. Write 0 to SCISPICKSREQ.
7. Poll until SCISPICKSRDY is read as 0.
8. When SCISPICKSRDY becomes 0, SCISPICLK starts to output. Clock switching is complete.

7) RA8M1
8) RA8D1
9) RA8E1
10) RA8T1
11) RA8E2

8.2.46 SCICKDIVCR : SCI Clock Division Control Register for RA8M1, RA8D1
8.2.42 SCICKDIVCR : SCI Clock Division Control Register for RA8T1, RA8E2
8.2.39 SCICKDIVCR : SCI Clock Division Control Register for RA8E1

Before Correction

SCICKDIV[2:0] bits (SCI Clock (SCICLK) Division Select)

These bits select the frequency of the SCI clock (SCICLK) and must be modified when SCICKCR.SCICKSRDY = 1.

After Correction

SCICKDIV[2:0] bits (SCI Clock (SCICLK) Division Select)

These bits select the frequency of the SCI clock (SCICLK) and must be modified when SCICKCR.SCICKSRDY = 1.

When switching the division ratio setting from n (n≠1), set all of MSTPCRB.MSTPBi(i=22,27,28,29,30,31) = 1 before setting SCICKCR.SCICKSREQ = 1.

8.2.48 SCICKCR : SCI Clock Control Register for RA8M1
8.2.49 SCICKCR : SCI Clock Control Register for RA8D1
8.2.44 SCICKCR : SCI Clock Control Register for RA8T1
8.2.41 SCICKCR : SCI Clock Control Register for RA8E1
8.2.45 SCICKCR : SCI Clock Control Register for RA8E2

Before Correction

The SCICKCR register controls the SCI clock.
When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of SCICKDIVCR.SCICKDIV[2:0] and SCICKSEL[2:0], use the following procedure:
1. Write 1 to SCICKSREQ.
2. Poll until SCICKSRDY is read as 1. While SCICKSRDY = 1, no clock is output to SCICLK.
3. Write to SCICKDIVCR. SCICKDIV[2:0] and SCICKSEL[2:0].
4. Write 0 to SCICKSREQ.
5. Poll until SCICKSRDY is read as 0.
6. When SCICKSRDY becomes 0, SCICLK starts to output. Clock switching is complete.

After Correction

The SCICKCR register controls the SCI clock.
When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of SCICKDIVCR. SCICKDIV[2:0] and SCICKSEL[2:0], use the following procedure:
1. Write 1 to all of MSTPCRB.MSTPBi (i=22,27,28,29,30,31) (only when switching the division ratio setting from n (n≠1))
2. Wait for two SCICLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to SCICKSREQ.
4. Poll until SCICKSRDY is read as 1. While SCICKSRDY = 1, no clock is output to SCICLK.
5. Write to SCICKDIVCR. SCICKDIV[2:0] and SCICKSEL[2:0].
6. Write 0 to SCICKSREQ.
7. Poll until SCICKSRDY is read as 0.
8. When SCICKSRDY becomes 0, SCICLK starts to output. Clock switching is complete.

7) RA8M1
8) RA8D1
9) RA8E1
10) RA8T1
11) RA8E2

8.2.47 SPICKDIVCR : SPI Clock Division Control Register for RA8M1,RA8D1

8.2.43 SPICKDIVCR : SPI Clock Division Control Register for RA8T1, RA8E2
8.2.40 SPICKDIVCR : SPI Clock Division Control Register for RA8E1

Before Correction

SPICKDIV[2:0] bits (SPI Clock (SPICLK) Division Select)

These bits select the frequency of the SPI clock (SPICLK) and must be modified when SPICKCR.SPICKSRDY = 1.

After Correction

SPICKDIV[2:0] bits (SPI Clock (SPICLK) Division Select)

These bits select the frequency of the SPI clock (SPICLK) and must be modified when SPICKCR.SPICKSRDY = 1.

When switching the division ratio setting from n (n≠1), set all of MSTPCRB.MSTPBi(i=18,19) = 1 before setting SPICKCR.SPICKSREQ = 1.

8.2.49 SPICKCR : SPI Clock Control Register for RA8M1
8.2.50 SPICKCR : SPI Clock Control Register for RA8D1
8.2.45 SPICKCR : SPI Clock Control Register for RA8T1
8.2.42 SPICKCR : SPI Clock Control Register for RA8E1
8.2.46 SPICKCR : SPI Clock Control Register for RA8E2

Before Correction

The SPICKCR register controls the SPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of SPICKDIVCR.SPICKDIV[2:0] and SPICKSEL[2:0], use the following procedure:

1. Write 1 to SPICKSREQ.
2. Poll until SPICKSRDY is read as 1. While SPICKSRDY = 1, no clock is output to SPICLK.
3. Write to SPICKDIVCR. SPICKDIV[2:0] and SPICKSEL[2:0].
4. Write 0 to SPICKSREQ.
5. Poll until SPICKSRDY is read as 0.
6. When SPICKSRDY becomes 0, SPICLK starts to output. Clock switching is complete.

After Correction

The SPICKCR register controls the SPI clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of SPICKDIVCR. SPICKDIV[2:0] and SPICKSEL[2:0], use the following procedure:

1. Write 1 to all of MSTPCRB.MSTPBi (i=18,19) (only when switching the division ratio setting from n (n≠1))
2. Wait for two SPICLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to SPICKSREQ.
4. Poll until SPICKSRDY is read as 1. While SPICKSRDY = 1, no clock is output to SPICLK.
5. Write to SPICKDIVCR. SPICKDIV[2:0] and SPICKSEL[2:0].

6. Write 0 to SPICKSREQ.
7. Poll until SPICKSRDY is read as 0.
8. When SPICKSRDY becomes 0, SPICLK starts to output. Clock switching is complete.

4) RA6T2

8.2.24 GPTCKDIVCR : GPT Clock Division Control Register for RA6T2

Before Correction

GPTCKDIV[2:0] bits (GPT Clock (GPTCLK) Division Select)

These bits select the frequency of the GPT clock (GPTCLK) and must be modified when GPTCKCR.GPTCKSRDY = 1.

After Correction

GPTCKDIV[2:0] bits (GPT Clock (GPTCLK) Division Select)

These bits select the frequency of the GPT clock (GPTCLK) and must be modified when GPTCKCR.GPTCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRE.MSTPE31 = 1 before setting GPTCKCR.GPTCKSREQ = 1.

8.2.28 GPTCKCR : GPT Clock Control Register for RA6T2

Before Correction

The GPTCKCR register controls the GPT clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of GPTCKDIVCR.GPTCKDIV[2:0] and GPTCKSEL[2:0], use the following procedure:

1. Write 1 to GPTCKSREQ.
2. Poll until GPTCKSRDY is read as 1. While GPTCKSRDY = 1, no clock is output to GPTCLK.
3. Write to GPTCKDIVCR. GPTCKDIV[2:0] and GPTCKSEL[2:0].
4. Write 0 to GPTCKSREQ.
5. Poll until GPTCKSRDY is read as 0.
6. When GPTCKSRDY becomes 0, GPTCLK starts to output. Clock switching is complete.

After Correction

The GPTCKCR register controls the GPT clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of GPTCKDIVCR. GPTCKDIV[2:0] and GPTCKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRE.MSTPE31 (only when switching the division ratio setting from n (n≠1))
2. Wait for two GPTCLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to GPTCKSREQ.
4. Poll until GPTCKSRDY is read as 1. While GPTCKSRDY = 1, no clock is output to GPTCLK.
5. Write to GPTCKDIVCR. GPTCKDIV[2:0] and GPTCKSEL[2:0].

6. Write 0 to GPTCKSREQ.
7. Poll until GPTCKSRDY is read as 0.
8. When GPTCKSRDY becomes 0, GPTCLK starts to output. Clock switching is complete.

8) RA8D1

11) RA8E2

8.2.48 LCDCKDIVCR : LCD Clock Division Control Register for RA8D1

8.2.44 LCDCKDIVCR : LCD Clock Division Control Register for RA8E2

Before Correction

LCDCKDIV[2:0] bits (LCD Clock (LCDCLK) Division Select)

These bits select the frequency of the LCD clock (LCDCLK) and must be modified when LCDCKCR.LCDCKSRDY = 1.

After Correction

LCDCKDIV[2:0] bits (LCD Clock (LCDCLK) Division Select)

These bits select the frequency of the LCD clock (LCDCLK) and must be modified when LCDCKCR.LCDCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRC.MSTPC4 = 1 before setting LCDCKCR.LCDCKSREQ = 1.

8.2.51 LCDCKCR : LCD Clock Control Register for RA8D1

8.2.47 LCDCKCR : LCD Clock Control Register for RA8E2

Before Correction

The LCDCKCR register controls the LCD clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of LCDCKDIVCR.LCDCKDIV[2:0] and LCDCKSEL[2:0], use the following procedure:

1. Write 1 to LCDCKSREQ.
2. Poll until LCDCKSRDY is read as 1. While LCDCKSRDY = 1, no clock is output to LCDCLK.
3. Write to LCDCKDIVCR. LCDCKDIV[2:0] and LCDCKSEL[2:0].
4. Write 0 to LCDCKSREQ.
5. Poll until LCDCKSRDY is read as 0.
6. When LCDCKSRDY becomes 0, LCDCLK starts to output. Clock switching is complete.

After Correction

The LCDCKCR register controls the LCD clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of LCDCKDIVCR. LCDCKDIV[2:0] and LCDCKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRC.MSTPC4 (only when switching the division ratio setting from n (n≠1))
2. Wait for two LCDCLK (only when switching the division ratio setting from n (n≠1))

3. Write 1 to LCDCKSREQ.
4. Poll until LCDCKSRDY is read as 1. While LCDCKSRDY = 1, no clock is output to LCDCLK.
5. Write to LCDCKDIVCR. LCDCKDIV[2:0] and LCDCKSEL[2:0].
6. Write 0 to LCDCKSREQ.
7. Poll until LCDCKSRDY is read as 0.
8. When LCDCKSRDY becomes 0, LCDCLK starts to output. Clock switching is complete.

4) RA6T2

8.2.25 IICCKDIVCR : IIC Clock Division Control Register for RA6T2

Before Correction

IICCKDIV[2:0] bits (IIC Clock (IICCLK) Division Select)

These bits select the frequency of the IIC clock (IICCLK) and must be modified when IICCKCR.IICCKSRDY = 1.

After Correction

IICCKDIV[2:0] bits (IIC Clock (IICCLK) Division Select)

These bits select the frequency of the IIC clock (IICCLK) and must be modified when IICCKCR.IICCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set all of MSTPCRB.MSTPBi(i=8,9) = 1 before setting IICCKCR.IICCKSREQ = 1.

8.2.29 IICCKCR : IIC Clock Control Register for RA6T2

Before Correction

The IICCKCR register controls the IIC clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of IICCKDIVCR.IICCKDIV[2:0] and IICCKSEL[2:0], use the following procedure:

1. Write 1 to IICCKSREQ.
2. Poll until IICCKSRDY is read as 1. While IICCKSRDY = 1, no clock is output to IICCLK.
3. Write to IICCKDIVCR. IICCKDIV[2:0] and IICCKSEL[2:0].
4. Write 0 to IICCKSREQ.
5. Poll until IICCKSRDY is read as 0.
6. When IICCKSRDY becomes 0, IICCLK starts to output. Clock switching is complete.

After Correction

The IICCKCR register controls the IIC clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of IICCKDIVCR. IICCKDIV[2:0] and IICCKSEL[2:0], use the following procedure:

1. Write 1 to all of MSTPCRB.MSTPBi(i=8,9) (only when switching the division ratio setting from n (n≠1))
2. Wait for two IICCLK (only when switching the division ratio setting from n (n≠1))

3. Write 1 to IICCKSREQ.
4. Poll until IICCKSRDY is read as 1. While IICCKSRDY = 1, no clock is output to IICCLK.
5. Write to IICCKDIVCR. IICCKDIV[2:0] and IICCKSEL[2:0].
6. Write 0 to IICCKSREQ.
7. Poll until IICCKSRDY is read as 0.
8. When IICCKSRDY becomes 0, IICCLK starts to output. Clock switching is complete.

6) RA6E2, RA4E2

8.2.26 CECCKDIVCR : CEC Clock Division Control Register for RA6E2, RA4E2

Before Correction

CECCKDIV[2:0] bits (CEC Clock (CECCLK) Division Select)

These bits select the frequency of the CEC clock (CECCLK) and must be modified when CECCKCR.CECCKSRDY = 1.

After Correction

CECCKDIV[2:0] bits (CEC Clock (CECCLK) Division Select)

These bits select the frequency of the CEC clock (CECCLK) and must be modified when CECCKCR.CECCKSRDY = 1.

When switching the division ratio setting from n (n≠1), set MSTPCRB.MSTPB3 = 1 before setting CECCKCR.CECCKSREQ = 1.

8.2.30 CECCKCR : CEC Clock Control Register for RA6E2, RA4E2

Before Correction

The CECCKCR register controls the CEC clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of CECCKDIVCR.CECCKDIV[2:0] and CECCKSEL[2:0], use the following procedure:

1. Write 1 to CECCKSREQ.
2. Poll until CECCKSRDY is read as 1. While CECCKSRDY = 1, no clock is output to CECCLK.
3. Write to CECCKDIVCR. CECCKDIV[2:0] and CECCKSEL[2:0].
4. Write 0 to CECCKSREQ.
5. Poll until CECCKSRDY is read as 0.
6. When CECCKSRDY becomes 0, CECCLK starts to output. Clock switching is complete.

After Correction

The CECCKCR register controls the CEC clock.

When switching the clock source, ensure that the clock before the switch and the clock after the switch generate stable output. To change the set value of CECCKDIVCR. CECCKDIV[2:0] and CECCKSEL[2:0], use the following procedure:

1. Write 1 to MSTPCRB.MSTPB3 (only when switching the division ratio setting from n (n≠1))

2. Wait for two CECCLK (only when switching the division ratio setting from n (n≠1))
3. Write 1 to CECCKSREQ.
4. Poll until CECCKSRDY is read as 1. While CECCKSRDY = 1, no clock is output to CECCLK.
5. Write to CECCKDIVCR. CECCKDIV[2:0] and CECCKSEL[2:0].
6. Write 0 to CECCKSREQ.
7. Poll until CECCKSRDY is read as 0.
8. When CECCKSRDY becomes 0, CECCLK starts to output. Clock switching is complete.

Reference document

Product	Document name
RA6M4 Group	Renesas RA6M4 Group User's Manual: Hardware Rev.1.40
RA6M5 Group	Renesas RA6M5 Group User's Manual: Hardware Rev.1.30
RA4M2 Group	Renesas RA4M2 Group User's Manual: Hardware Rev. 1.30
RA4M3 Group	Renesas RA4M3 Group User's Manual: Hardware Rev. 1.40
RA6E1 Group	Renesas RA6E1 Group User's Manual: Hardware Rev.1.20
RA4E1 Group	Renesas RA4E1 Group User's Manual: Hardware Rev. 1.20
RA6T2 Group	Renesas RA6T2 Group User's Manual: Hardware Rev. 1.40
RA6T3 Group	Renesas RA6T3 Group User's Manual: Hardware Rev. 1.20
RA6E2 Group	Renesas RA6E2 Group User's Manual: Hardware Rev. 1.30
RA4E2 Group	Renesas RA4E2 Group User's Manual: Hardware Rev. 1.30
RA8M1 Group	Renesas RA8M1 Group User's Manual: Hardware Rev. 1.20
RA8D1 Group	Renesas RA8D1 Group User's Manual: Hardware Rev. 1.20
RA8E1 Group	Renesas RA8E1 Group User's Manual: Hardware Rev. 1.00
RA8T1 Group	Renesas RA8T1 Group User's Manual: Hardware Rev. 1.20
RA8E2 Group	Renesas RA8E2 Group User's Manual: Hardware Rev. 1.00