Date: Sep. 29, 2014

RENESAS TECHNICAL UPDATE

1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan
Renesas Electronics Corporation

Product Category	MPU & MCU		Document No.	TN-RX*-A106A/E	Rev.	1.00
Title	Corrections to Manual regarding the USB 2.0 Host/ Function Module (USBc) in the RX111 Group		Information Category	Technical Notification		
Applicable Product	RX111 Group	Lot No. All	Reference Document	RX111 Group User's Manual: Hardware Rev.1.10 (R01UH0365EJ0110)		

This document describes corrections to section 25, USB 2.0 Host/Function Module (USBc) in RX111 Group User's Manual: Hardware.

•Overall

The following function name is corrected:

Before correction

SOF interpolation function

After correction

SOF recovery function

•Page 642 of 1243

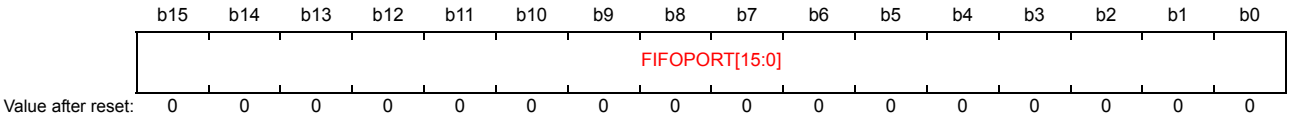
The following note for the DVSTCTR0.WKUP bit in 25.2.3 is deleted:
Note 1. Only 1 can be written.

• Pages 645, 646 of 1243

The descriptions for registers CFIFO, D0FIFO, D1FIFO in 25.2.4 are corrected as follows:

Before correction

Address(es): CFIFO 000A 0014h, D0FIFO 000A 0018h, D1FIFO 000A 001Ch



Bit	Symbol	Bit Name	Description	R/W
b15 to b0	FIFOPORT[15:0]	FIFO Port	The valid bits in a FIFO port register depend on the settings of the corresponding MBW (CFIFOSEL.MBW, D0FIFOSEL.MBW, and D1FIFOSEL.MBW) and BIGEND bits (CFIFOSEL.BIGEND, D0FIFOSEL.BIGEND, and D1FIFOSEL.BIGEND) as shown in Table 25.5 and Table 25.6	R/W

FIFOPORT[15:0] Bits (FIFO Port)

Accessing the FIFOPORT[15:0] bits allow reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer.

Each FIFO port register can be accessed only while the FRDY bit in each port control register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL) is 1.

The valid bits in a FIFO port register depend on the settings of the corresponding MBW and BIGEND bits in a port select register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL) as shown in Table 25.5 and Table 25.6

Table 25.5 Endian Operation in 16-Bit Access

CFIFOSEL.BIGEND Bit D0FIFOSEL.BIGEND Bit D1FIFOSEL.BIGEND Bit	Bits 15 to 8	Bits 7 to 0
0	N + 1 data	N + 0 data
1	N + 0 data	N + 1 data

Table 25.6 Endian Operation in 8-Bit Access

CFIFOSEL.BIGEND Bit D0FIFOSEL.BIGEND Bit D1FIFOSEL.BIGEND Bit	Bits 15 to 8	Bits 7 to 0
0	Access prohibited *1	N + 0 data
1	Access prohibited *1	N + 0 data

Note 1. Reading from an access-prohibited area is not allowed.

After correction

(1) When the MBW bit is 1

Address(es): CFIFO 000A 0014h, D0FIFO 000A 0018h, D1FIFO 000A 001Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	FIFO Port	This port is used for reading receive data from the FIFO buffer and writing transmit data to the FIFO buffer.	R/W

(2) When the MBW bit is 0

Address(es): CFIFO 000A 0014h, D0FIFO 000A 0018h, D1FIFO 000A 001Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	L[7:0]							
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	L[7:0]	FIFO Port	This port is used for reading receive data from the FIFO buffer and writing transmit data to the FIFO buffer.	R/W

FIFO Port Bits

Accessing the FIFO port bits allows reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer.

Each FIFO port register can be accessed only while the FRDY bit in each FIFO port control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR) is 1.

The valid bits in a FIFO port register depend on the settings of the corresponding MBW bit of the FIFO port select register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL).

When the MBW bit is 1 (16-bit width), the data arrangement may differ from the data arrangement on the RAM depending on the value of the MDE.MDE[2:0] bits and the setting of the BIGEND bit (CFIFOSEL.BIGEND, D0FIFOSEL.BIGEND, or D1FIFOSEL.BIGEND). Table 25.5 lists the endian operation in 16-bit access. Note that if the total number of transmit data bytes is odd, access the L[7:0] bits in bytes when writing the last data.

When the MBW bit is 0 (8-bit width), access the L[7:0] bits in bytes.

Table 25.5 Endian Operation in 16-Bit Access

MDE.MDE[2:0] bits	CFIFOSEL.BIGEND Bit D0FIFOSEL.BIGEND Bit D1FIFOSEL.BIGEND Bit	Bits 15 to 8	Bits 7 to 0	Remarks
000b (big endian)	0 (little endian)	Data in address N + 1	Data in address N	Bytes reversed
	1 (big endian)	Data in address N	Data in address N + 1	
111b (little endian)	0 (little endian)	Data in address N + 1	Data in address N	
	1 (big endian)	Data in address N	Data in address N + 1	Bytes reversed

•Page 651 of 1243

The note for the BCLR bit in registers CFIFOCTR, D0FIFOCTR, and D1FIFOCTR in 25.2.6 is added as follows:

Note 1. This bit is read as 0.

•Page 651 of 1243

The following note for the BVAL bit in registers CFIFOCTR, D0FIFOCTR, and D1FIFOCTR in 25.2.6 is deleted:

Note 1. Only 1 can be written.

•Page 659 of 1243

The notes for the value after reset of the INTSTS0 register in 25.2.13 are corrected as follows:

Before correction

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
Value after reset:	0	0	0	0/1 *1	0	0	0	0	0 *2	0	0	0/1 *3	0	0	0	0

Note 1. The value is 0 after a power-on reset and 1 after a USB bus reset.
Note 2. The value is 1 when the USB0_VBUS pin is high and 0 when the USB0_VBUS pin is low.
Note 3. The value is 000b after a power-on reset and 001b after a USB bus reset.

After correction

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
Value after reset:	0	0	0	0/1 *1	0	0	0	0	0 *2	0 *3	0 *3	0/1 *3	0	0	0	0

Note 1. The value is 0 after the MCU is reset, and the value is 1 after a USB bus reset.
Note 2. The value is 1 when the USB0_VBUS pin is high, and the value is 0 when the USB0_VBUS pin is low.
Note 3. The value is 000b after the MCU is reset, and the value is 001b after a USB bus reset.

•Page 668 of 1243

The note for bits CRCE and OVRN of the FRMNUM register in 25.2.18 is corrected as follows:

Before correction

Note 1. Only 0 can be written.

After correction

Note 1: When setting each status to 0, write 0 to the bit that is cleared, and write 1 to the other bit.

•Page 668 of 1243

The descriptions for the CRCE bit in 25.2.18 is corrected as follows:

Before correction

(1) When the host controller function is selected
On detecting a CRC error, the USB generates the internal NRDY interrupt request.

After correction

The USB generates an internal NRDY interrupt request when a CRC error is detected.

•Page 675 of 1243

The note for bits SQSET and SQCLR of the DCPCTR register in 25.2.25 is corrected as follows:

Before correction

Note 1. This bit is read as 0. Only 1 can be written.

After correction

Note 1. This bit is read as 0.

•Page 675 of 1243

The following note for bits SUREQCLR and SUREQ of the DCPCTR register in 25.2.25 is deleted:
Note 2. Only 1 can be written.

•Page 681 of 1243

Descriptions for the PIPEMAXP.MXPS[8:0] bits in 25.2.28 are corrected as follows:

Before correction

Specifies the maximum data payload (maximum packet size) for the selected pipe.
These bits should be set to the appropriate value for each transfer type based on USB Specifications 2.0. While MXPS[8:0] = 0, do not write to the FIFO buffer or set PID to BUF.

After correction

Specifies the maximum data payload (maximum packet size) for the selected pipe.
These bits should be set to the appropriate value for each transfer type based on USB Specifications 2.0. Note that the maximum value for PIPE1 and PIPE2 is 256. While the MXPS[8:0] bits are 000h, do not write to the FIFO buffer or do not set the PID[1:0] bits to 01b (BUF).

•Pages 683, 688 of 1243

The note for bits SQSET and SQCLR of the PIPEnCTR register in 25.2.30 is corrected as follows:

Before correction

Note 1. Only 0 can be read. Only 1 can be written.

After correction

Note 1. This bit is read as 0.

•Page 697 of 1243

The descriptions in 25.3.1.2 are corrected as follows:

Before correction

For the USB, the host or function controller can be selected using the DCFM bit in SYSCFG. The DCFM bit should be modified in the initial settings immediately after a power-on reset or in the D+ pull-up-disabled (SYSCFG.DPRPU bit = 0) and D+ and D– pull-down-disabled (SYSCFG.DRPD bit = 0) state.

After correction

For the USB0, the host controller function or function controller function can be selected using the SYSCFG.DCFM bit. Set the DCFM bit during initialization after a reset is released or while D+ pull-up and D+/D– pull-down are disabled (bits SYSCFG.DPRPU and SYSCFG.DRPD are 0).

•Page 708 of 1243

Figure 25.8 is corrected as follows:

Before correction

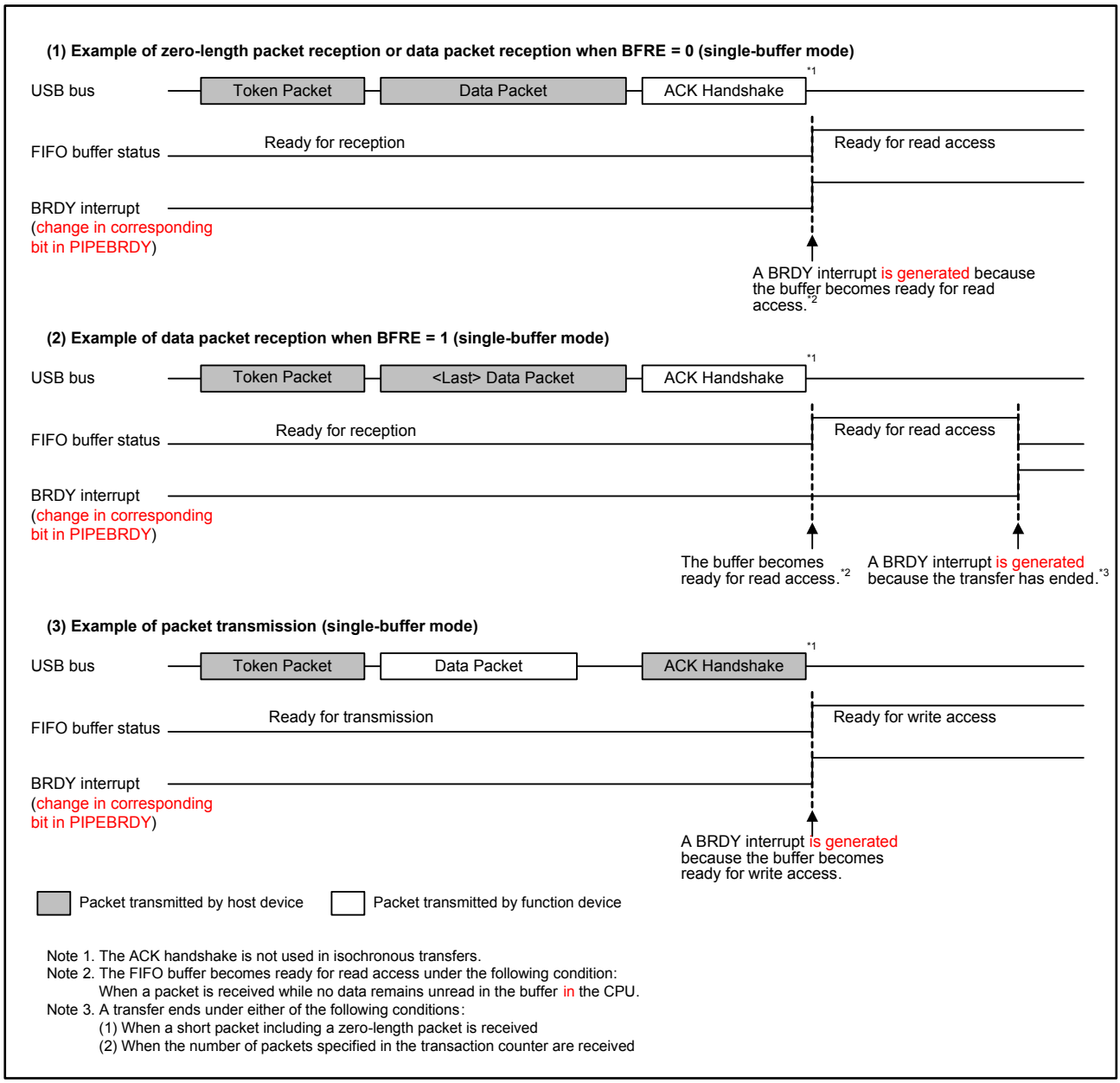
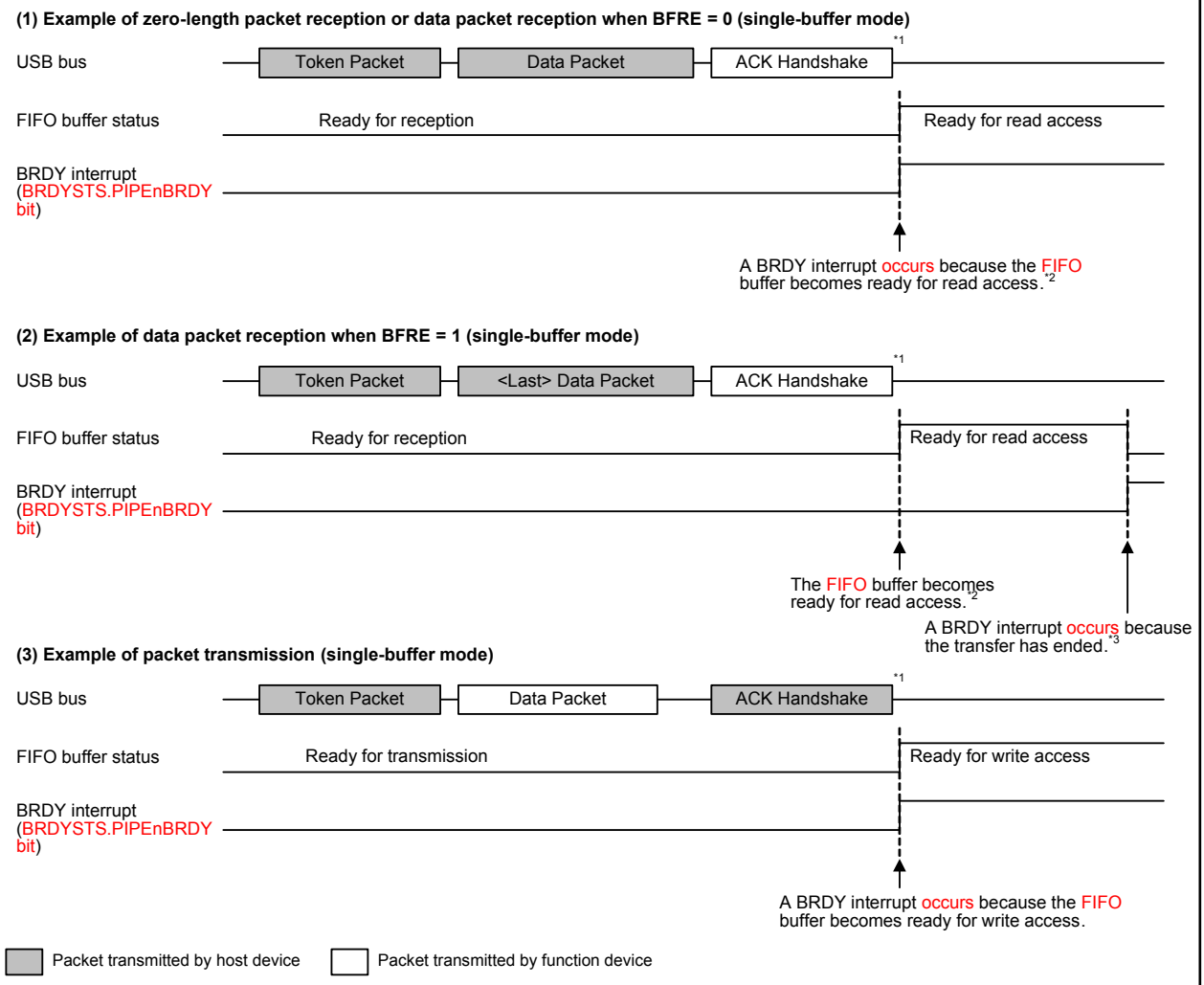


Figure 25.8 Timing of BRDY Interrupt Generation

After correction



Note 1. The ACK handshake is not used in isochronous transfers.
Note 2. The FIFO buffer becomes ready for read access under the following condition:
When a packet is received while no data remains unread in the FIFO buffer on the CPU side.
Note 3. A transfer ends under either of the following conditions:
(1) When a short packet including a zero-length packet is received
(2) When the number of packets specified in the transaction counter are received

Figure 25.8 Timing of BRDY Interrupt Generation

•Page 711 of 1243

Figure 25.9 is corrected as follows:

Before correction

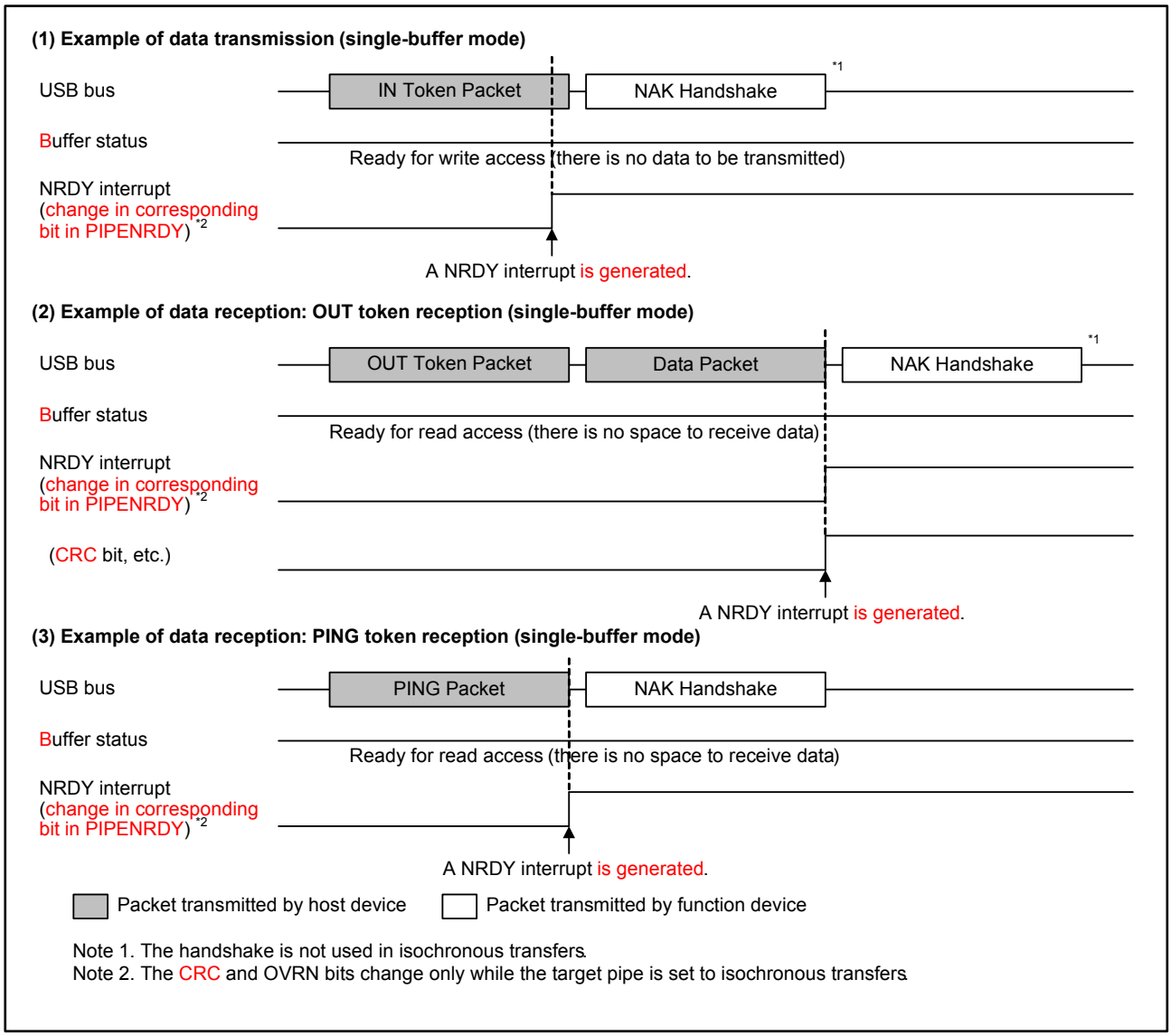


Figure 25.9 Timing of NRDY Interrupt Generation (When Function Controller is Selected)

After correction

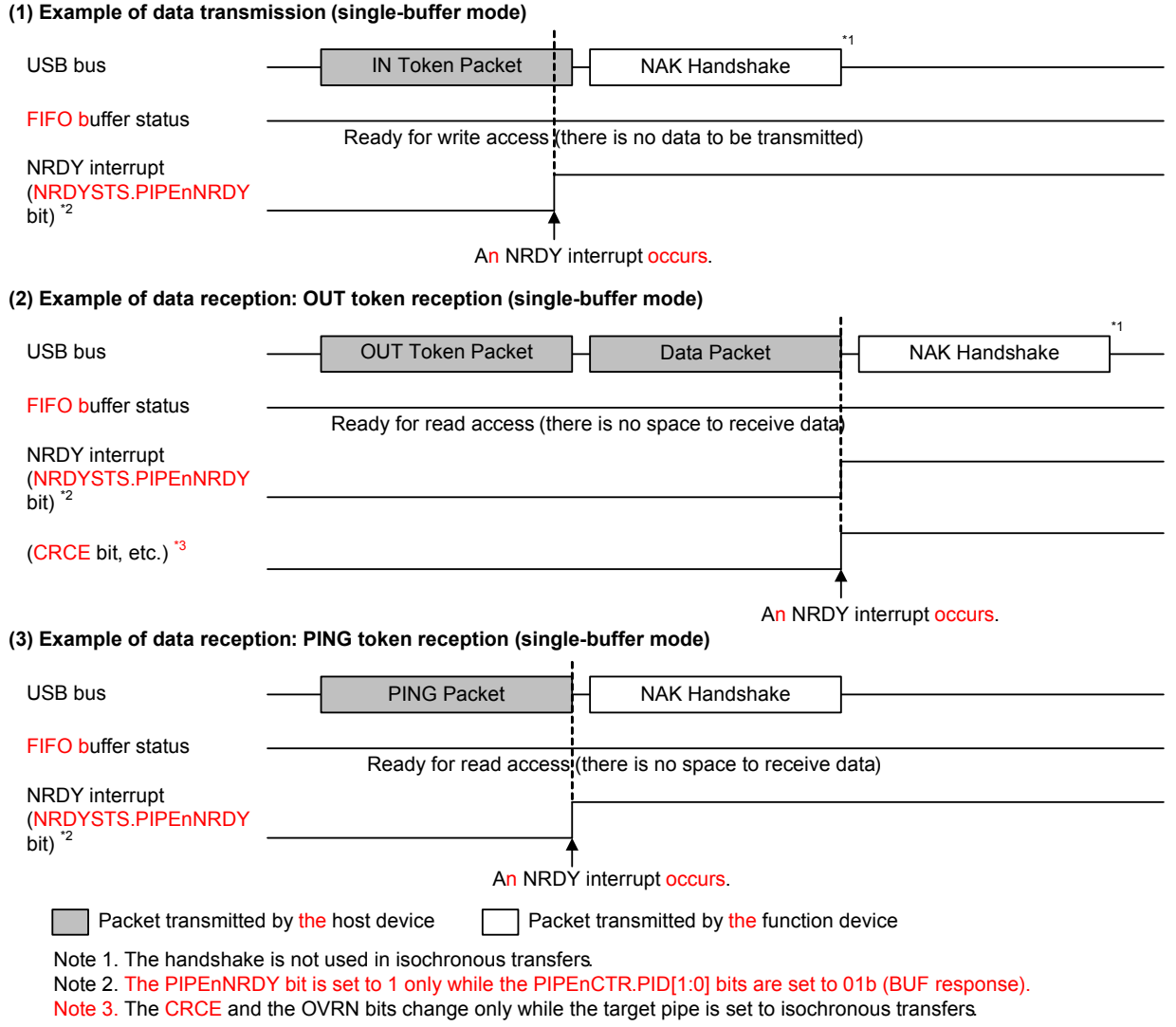


Figure 25.9 Timing of NRDY Interrupt Generation (When Function Controller Function is Selected)

•Page 712 of 1243

Figure 25.10 is corrected as follows:

Before correction

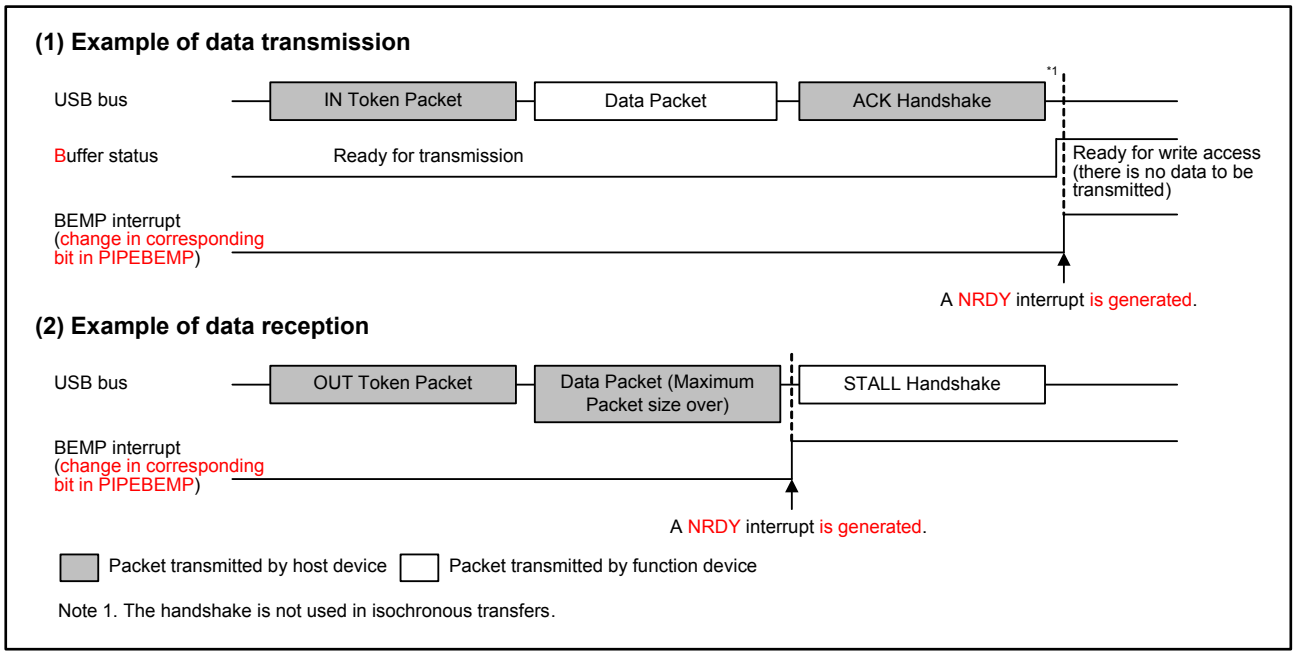


Figure 25.10 Timing of BEMP Interrupt Generation (When Function Controller is Selected)

After correction

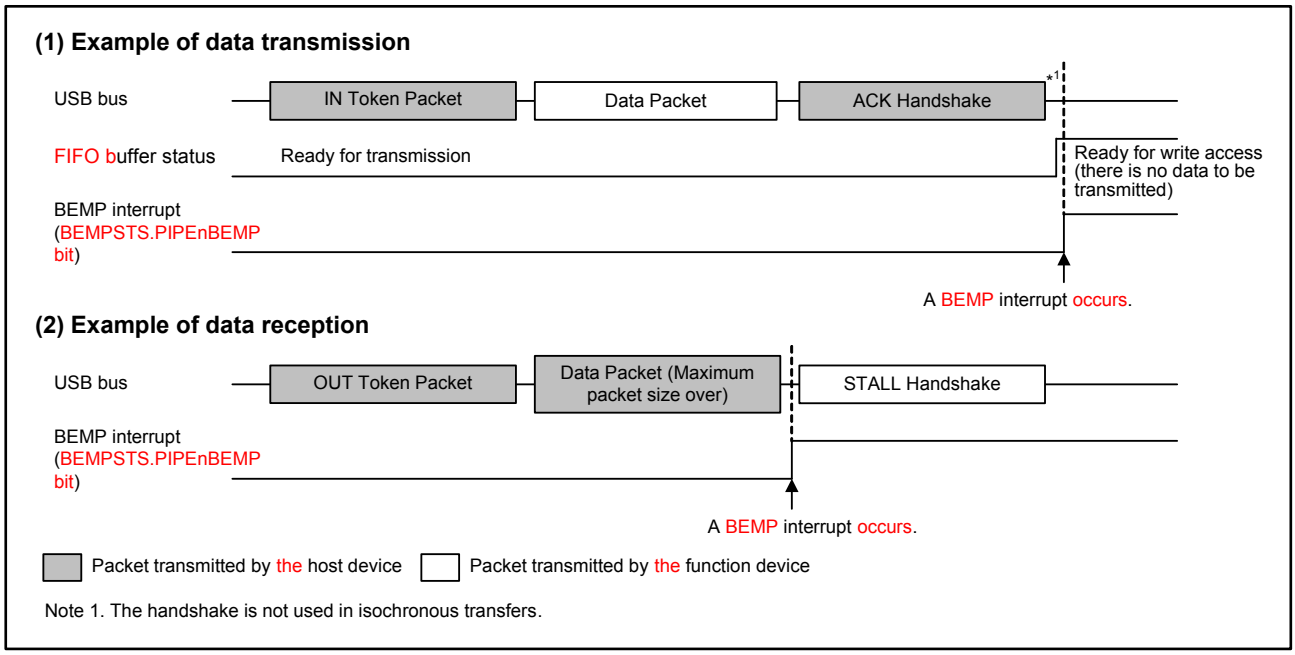


Figure 25.10 Timing of BEMP Interrupt Generation (When Function Controller Function is Selected)

•Page 726 of 1243

The descriptions for (4) Control Transfer Auto Response Function in 25.3.6.2 are added as follows:

(4) Control Transfer Auto Response Function

The USB automatically responds to a correct SET_ADDRESS request. If any of the following errors occurs in the SET_ADDRESS request, a response from the software is necessary.

- bmRequestType is not 00h: Any transfer other than a control write transfer
- wIndex is not 00h: Request error
- wLength is not 00h: Any transfer other than a no-data control transfer
- wValue is larger than 7Fh: Request error
- The INTSTS0.DVSQ[2:0] bits are 011b (configured state): Control transfer of a device state error

For all requests other than the SET_ADDRESS request, a response is required from the corresponding software.

•Page 727 of 1243

The descriptions for (1) Counter Initialization in 25.3.8.1 are corrected as follows:

Before correction

(1) Counter Initialization

The USB controller initializes the interval counter under the following conditions.

- Power-on reset:
  The IITV[2:0] bits are initialized.
- Buffer memory initialization using the PIPEnCTR.ACLRM bit:
  The IITV[2:0] bits are not initialized but the count value is initialized. Setting the PIPEnCTR.ACLRM bit to 0 starts counting from the value set in the IITV bits.

After correction

(1) Counter Initialization

The interval counter is initialized when the MCU is reset or when the PIPEnCTR.ACLRM bit is set to 1. The PIPEPERI.IITV[2:0] bits are not initialized when the interval counter is initialized by using the ACLRM bit.

•Page 730 of 1243

The descriptions for (1) Counter Initialization when Function Controller is Selected in 25.3.9.3 (1) are corrected as follows:

Before correction

(1) Counter Initialization when Function Controller is Selected

The USB initializes the interval counter under the following conditions.

- Power-on Reset
  The PIPEPERI.IITV[2:0] bits are initialized.
- Buffer memory initialization using the ACLRM bit
  The IITV[2:0] bits are not initialized but the count value is initialized.

After correction

(1) Counter Initialization

The interval counter is initialized when the MCU is reset or when the PIPEnCTR.ACLRM bit is set to 1. The PIPEPERI.IITV[2:0] bits are not initialized when the interval counter is initialized by using the ACLRM bit.

•Page 736 of 1243

The descriptions for SOF Interpolation Function in 25.3.10 are corrected as follows:

Before correction

25.3.10 SOF Interpolation Function

When the function controller is selected and if data could not be received at intervals of 1 ms because an SOF packet was corrupted or missing, the USB interpolates the SOF. The SOF interpolation operation begins when the USBE and SCKE bits in SYSCFG have been set to 1 and an SOF packet is received. The interpolation function is initialized under the following conditions.

- Power-on reset
- USB bus reset
- Suspended state detected

The SOF interpolation operates as follows.

- The interpolation function is not activated until an SOF packet is received.
- After the first SOF packet is received, interpolation is carried out by counting 1 ms with an internal clock of 48 MHz.
- After the second and subsequent SOF packets are received, interpolation is carried out at the previous reception interval.
- Interpolation is not carried out in the suspended state or while a USB bus reset is being received.

The USB supports the following functions based on the SOF packet reception. These functions also operate normally with SOF interpolation, if the SOF packet was missing.

- Updating of the frame number
- SOFR interrupt timing
- Isochronous transfer interval count

If an SOF packet is missing, the FRMNUM.FRNM[10:0] bits are not updated.

After correction

25.3.10 SOF Recovery Function

When the function controller is selected and if data could not be received at intervals of 1 ms because an SOF packet was corrupted or missing, the USB recovers the SOF. The SOF recovery operation begins when the USBE and SCKE bits in SYSCFG have been set to 1 and an SOF packet is received. The recovery function is initialized under the following conditions.

- MCU reset
- USB bus reset
- Suspended state detected

The SOF recovery operates as follows.

- The recovery function is not activated until an SOF packet is received.
- After the first SOF packet is received, recovery is carried out by counting 1 ms with an internal clock of 48 MHz.
- After the second and subsequent SOF packets are received, recovery is carried out at the previous reception interval.
- Recovery is not carried out in the suspended state or while a USB bus reset is being received.

The USB supports the following functions based on the SOF packet reception. These functions also operate normally with SOF recovery, if the SOF packet was missing.

- Updating of the frame number
- SOFR interrupt timing
- Isochronous transfer interval count

If an SOF packet is missing, the FRMNUM.FRNM[10:0] bits are not updated.