Date: Jul. 4, 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*-A102A/E	Rev.	1.00
Title	Corrections to Manual regarding the USB 2.0 Host/ Function Module (USB) in the RX62N Group and RX621 Group		Information Category	Technical Notification		
Applicable Product	RX62N Group, RX621 Group	Lot No. All	Reference Document	RX62N Group, RX621 Group User's Manual: Hardware Rev.1.30 (R01UH0033EJ0130)		

This document describes corrections to section 28, USB 2.0 Host/Function Module (USB) in RX62N Group, RX621 Group User's Manual: Hardware.

•Overall

The following function name is corrected:

Before correction
SOF interpolation function

After correction
SOF recovery function

•Page 1269 of 2020

Note 1 is added to Table 28.1 as follows:

Item	Specifications
Features	(1) Features of the USB host controller • Full-speed transfer (12 Mbps) is supported. *1 • Communications with multiple peripheral devices connected via a single HUB • Automatic scheduling for SOF and packet transmissions • Programmable intervals for isochronous and interrupt transfers (2) Features of the USB function controller • Full-speed transfer (12 Mbps) is supported. *1 • Control transfer stage control function • Device state control function • Auto response function for SET_ADDRESS request • SOF recovery function

Note 1. Low-speed transfer (1.5 Mbps) is not supported.

•Page 1279 of 2020

The description column for DVSTCTR0.RHST[2:0] bits in 28.2.3 is corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description
b2 to b0	RHST[2:0]	USB Bus Reset Status	• When the host controller function is selected b2 b1 b0 0 0 0: Communication speed not determined (powered state or no connection) 1 x x: USB bus reset in progress 0 0 1: Low-speed connection*1 0 1 0: Full-speed connection [Legend] x: Don't care • When the function controller function is selected b2 b1 b0 0 0 0: Communication speed not determined 1 0 0: USB bus reset in progress 0 1 0: Full-speed connection

After correction

Bit	Symbol	Bit Name	Description
b2 to b0	RHST[2:0]	USB Bus Reset Status	• When the host controller function is selected b2 b1 b0 0 0 0: Communication speed not determined (powered state or no connection) 1 x x: USB bus reset in progress 0 0 1: Low-speed connection *1 0 1 0: Full-speed connection [Legend] x: Don't care • When the function controller function is selected b2 b1 b0 0 0 0: Communication speed not determined 0 1 0: USB bus reset in progress or full-speed connection

•Page 1279 of 2020

The following note for the DVSTCTR0.WKUP bit in 28.2.3 is deleted:

Note 2. Only 1 can be written.

•Page 1280 of 2020

The following description for the DVSTCTR0.RESUME bit in 28.2.3 is corrected as follows:

Before correction

The RESUME bit controls the resume signal output when the host controller function is selected.

Setting the RESUME bit to 1 allows the USB module to drive the port to the K-state and output the resume signal.

— omitted —

After correction

The RESUME bit controls the resume signal output when the host controller function is selected.

Setting the RESUME bit to 1 allows the USB module to drive the port to the K-state and output the resume signal. When a remote wakeup signal is detected while the RWUPE bit is 1 in the suspended state, the USB module sets the RESUME bit to 1 and performs the same operation.

— omitted —

•Pages 1282, 1283 of 2020

The descriptions for registers CFIFO, D0FIFO, D1FIFO in 28.2.4 are corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	L[7:0]	FIFO Port	The valid bits in a FIFO port register depend on the settings of the corresponding MBW and BIGEND bits as shown in Table 28.7 and Table 28.8	R/W
b7 to b0	H[7:0]			

L[7:0]/H[7:0] (FIFO Port)

Accessing the FIFOPORT 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 control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR) is 1.

The valid bits in a FIFO port register depend on the settings of the corresponding MBW and BIGEND bits as shown in Table 28.7 and Table 28.8.

In 8-bit access, this bit should be accessed in bytes. The BIGEND bit setting is disabled.

In 16-bit access, this bit should be accessed in words. If a total number of data is odd, the last data should be accessed in bytes.

In both 8-bit access and 16-bit access, the FIFO register start address should be accessed regardless of settings.

Table 28.7 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 28.8 Endian Operation in 8-Bit Access

CFIFOSEL.BIGEND Bit D0FIFOSEL.BIGEND Bit D1FIFOSEL.BIGEND Bit	Bits 15 to 8	Bits 7 to 0
x (Setting is invalid)	Access prohibited *	N + 0 data

Note: * Accessing an access-prohibited area to read data is not allowed.

After correction

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	L[7:0]	FIFO Port	The valid bits in a FIFO port register depend on the settings of the corresponding MBW bit. When the MBW bit is 1 (16-bit width), the data arrangement varies depending on the state of the MDE pin and the BIGEND bit setting. See Table 28.7 for details. When the MBW bit is 0 (8-bit width), access the beginning address in bytes.	R/W
b7 to b0	H[7:0]			

L[7:0]/H[7:0] (FIFO Port)

Accessing the FIFO port 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 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 (CFIFOSEL.MBW, D0FIFOSEL.MBW, or D1FIFOSEL.MBW). When the MBW bit is 1 (16-bit width), the data arrangement may differ from the data arrangement on the RAM depending on the state of the MDE pin and the setting of the BIGEND bit (CFIFOSEL.BIGEND, D0FIFOSEL.BIGEND, or D1FIFOSEL.BIGEND). Table 28.7 lists the endian operation in 16-bit access.
In 8-bit access, these bits should be accessed in bytes. The BIGEND bit setting is disabled.
In 16-bit access, these bits should be accessed in words. Note that if the total number of transmit data bytes is odd, access in bytes when writing the last data.
In both 8-bit access and 16-bit access, the FIFO register beginning address should be accessed regardless of settings.

Table 28.7 Endian Operation in 16-Bit Access

MDMONR.MDE flag	CFIFOSEL.BIGEND Bit D0FIFOSEL.BIGEND Bit D1FIFOSEL.BIGEND Bit	Bits 15 to 8	Bits 7 to 0	Remarks
0 (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
1 (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	

•Page 1284 of 2020
The note for the REW bit in registers CFIFOSEL, D0FIFOSEL, D1FIFOSEL in 28.2.5 is corrected as follows:

Before correction
Note : * Only 0 can be read.

After correction
Note : * This bit is read as 0.

•Page 1289 of 2020
The note for the BCLR bit in registers CFIFOCTR, D0FIFOCTR, and D1FIFOCTR in 28.2.6 is corrected as follows:

Before correction
Note 1. Only 0 can be read and 1 can be written.

After correction
Note 1. This bit is read as 0.

•Page 1289 of 2020
The following note for the BVAL bit in registers CFIFOCTR, D0FIFOCTR, and D1FIFOCTR in 28.2.6 is deleted:
Note 2. Only 1 can be written.

•Page 1298 of 2020

The SOFCFG.TRNENSELbit in 28.2.12 is corrected as follows:

Before correction

Bit	Symbol	Bit Name	Description
b8	TRNENSEL	Transaction-Enabled Time Select	0: For non-low-speed communication 1: Setting prohibited

After correction

Bit	Symbol	Bit Name	Description
b8	—	Reserved	This bit is always read as 0. The write value should always be 0.

•Page 1298 of 2020

The following descriptions for SOFCFG.TRNENSELbit in 28.2.12 are deleted:

TRNENSEL Bit (Transaction-Enabled Time Select)

The TRNENSEL bit selects, for full-speed or low-speed communication, the transaction-enabled time in which the USB module issues tokens in a frame via the port.
The TRNENSEL bit is valid only when the host controller function is selected.
This bit should be set to 0 if the function controller function is selected.

•Page 1300 of 2020

The notes for the value after reset of the INTSTS0 register in 28.2.13 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 *3	0	0	0/1 *2	0	0	0	0

Note 1. This bit is initialized to 0b by a power-on reset and 1b by a USB bus reset.
Note 2. These bits are initialized to 000b by a power-on reset and 001b by a USB bus reset.
Note 3. This bit is initialized to 1 when the level of the USBm_VBUS pin input is high and 1 when low.

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 USBm_VBUS pin is high, and the value is 0 when the USBm_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 1308 of 2020

The note for bits CRCE and OVRN of the FRMNUM register in 28.2.18 is corrected as follows:

Before correction

Note : * Only 0 can be written.

After correction

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

•Page 1308 of 2020

The descriptions for the CRCE bit in 28.2.18 is corrected as follows:

Before correction

(1) When the host controller function is selected
On detecting a CRC error, the USB module generates the internal NRDY interrupt request.
(2) When the function controller function is selected
On detecting a CRC error, the USB module does not generate the internal NRDY interrupt request.

After correction

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

•Page 1316 of 2020

The note for bits SQSET and SQCLR of the DCPCTR register in 28.2.27 is corrected as follows:

Before correction

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

After correction

Note 1. This bit is read as 0.

•Page 1316 of 2020

Note symbol "*1" is deleted from the R/W column of the DCPCTR.SUREQCLR bit in 28.2.27.

•Page 1316 of 2020

The following note for the DCPCTR.SUREQ bit in 28.2.27 is deleted:
Note 2. Only 1 can be written.

•Page 1323 of 2020

Descriptions for the PIPEMAXP.MSPS[8:0] bits in 30.2.30 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. While MXPS = 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. 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 1325, 1330 of 2020

The note for bits SQSET and SQCLR of the PIPEnCTR register in 28.2.32 is corrected as follows:

Before correction

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

After correction

Note 1. This bit is read as 0.

•Page 1342 of 2020

The descriptions in 28.3.1.2 are corrected as follows:

Before correction

For the USB module, the host controller function or function controller function 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 (DPRPU = 0) and D+ and D- pull-down-disabled (DRPD = 0) state.

After correction

For the USB module, 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 1357 of 2020

Figure 28.11 is corrected as follows:

Before correction

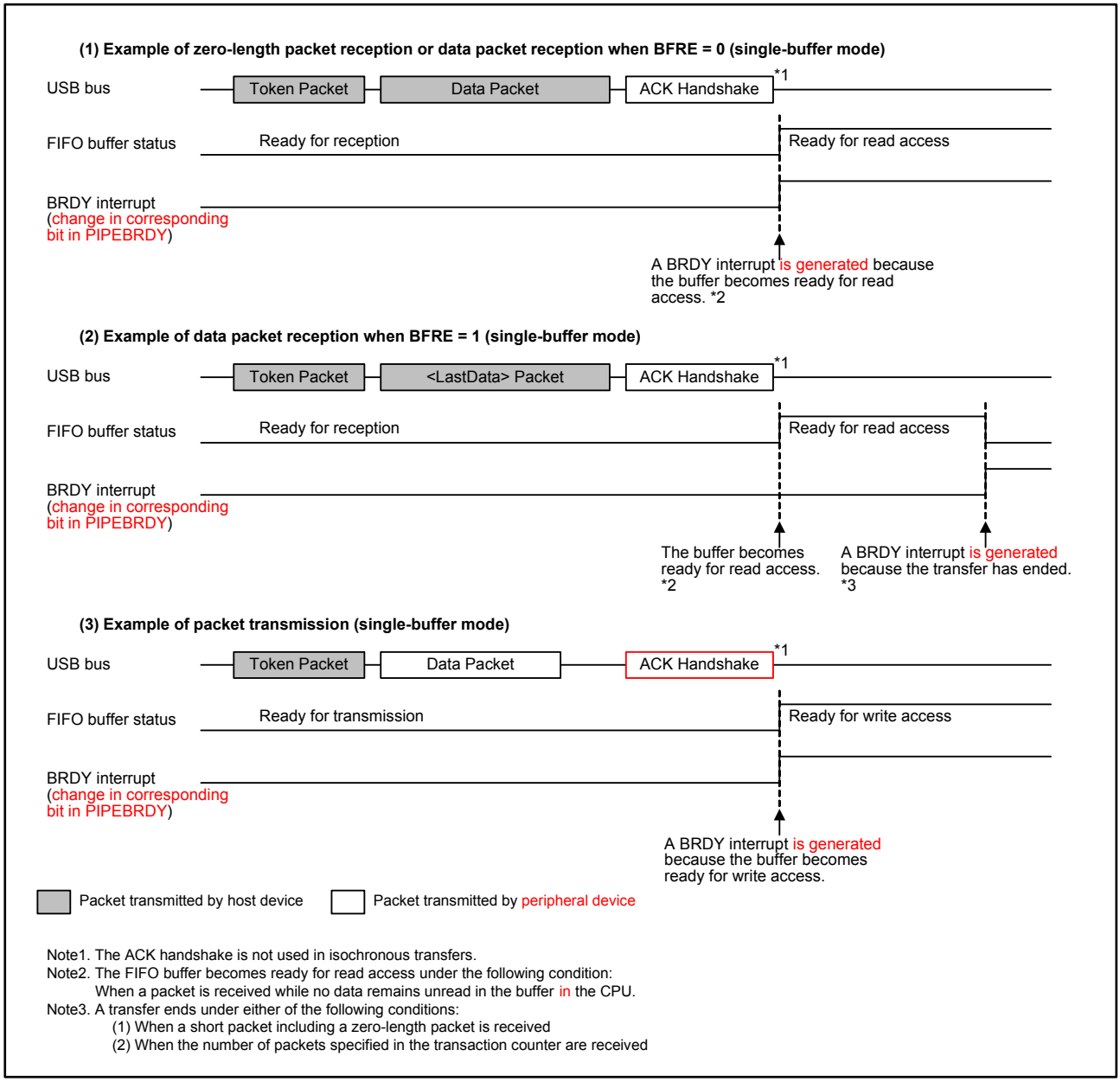
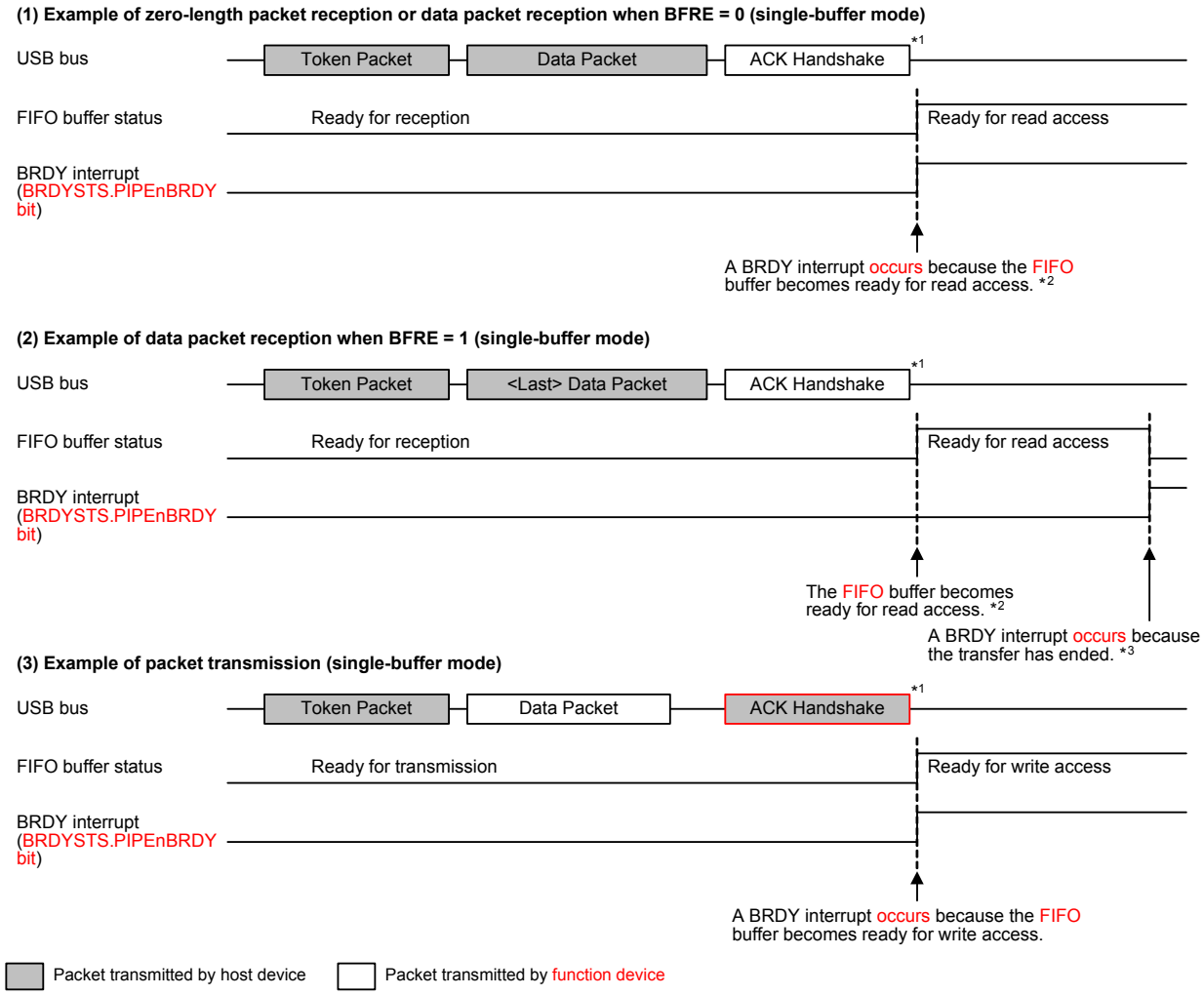


Figure 28.11 Timing of BRDY Interrupt Generation (When Function Controller Function is Selected)

After correction



Note1. The ACK handshake is not used in isochronous transfers.
Note2. 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.
Note3. 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 28.11 Timing of BRDY Interrupt Generation (When Function Controller Function is Selected)

•Page 1360 of 2020

Figure 28.12 is corrected as follows:

Before correction

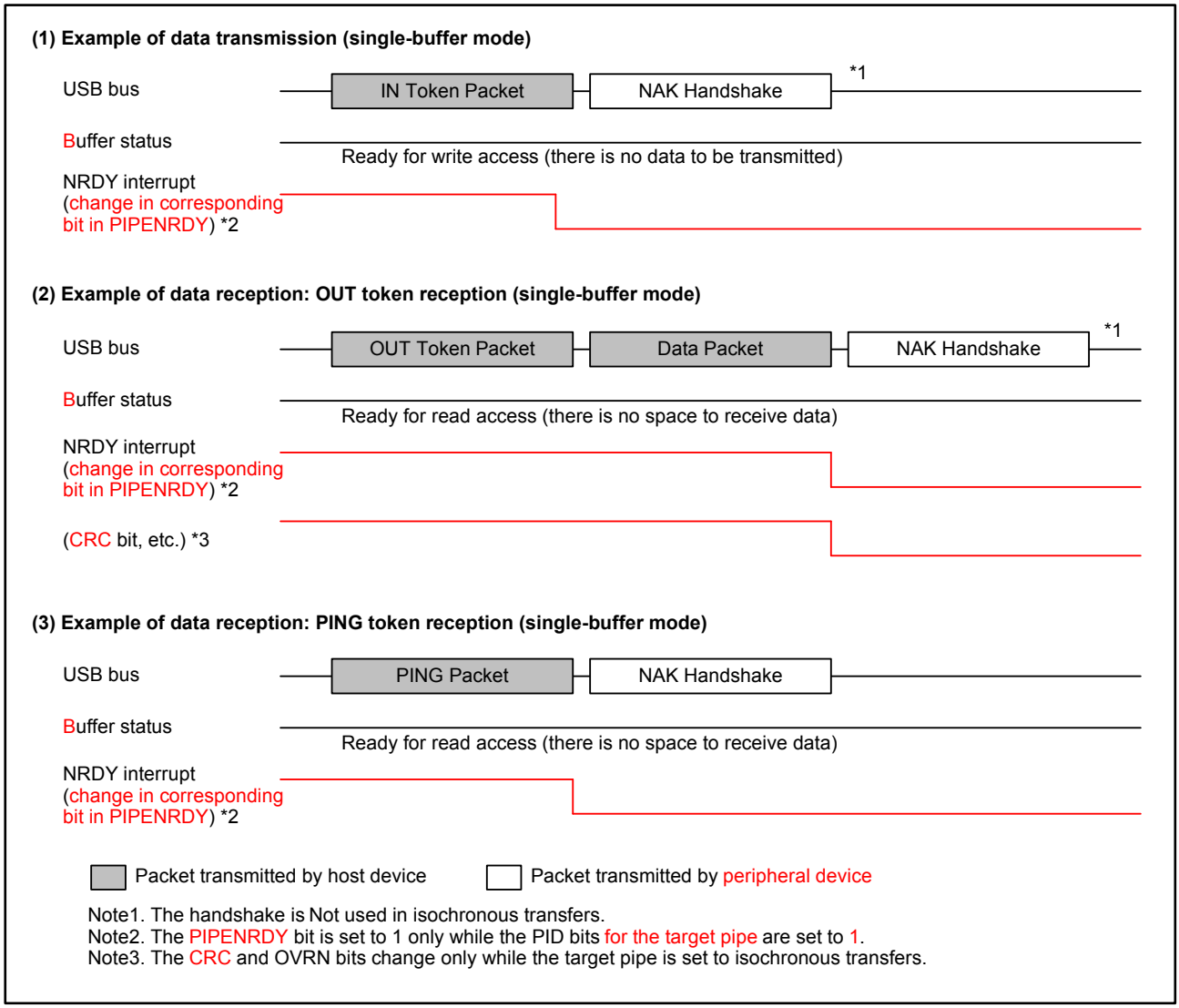


Figure 28.12 Timing of NRDY Interrupt Generation (When Function Controller Function is Selected)

After correction

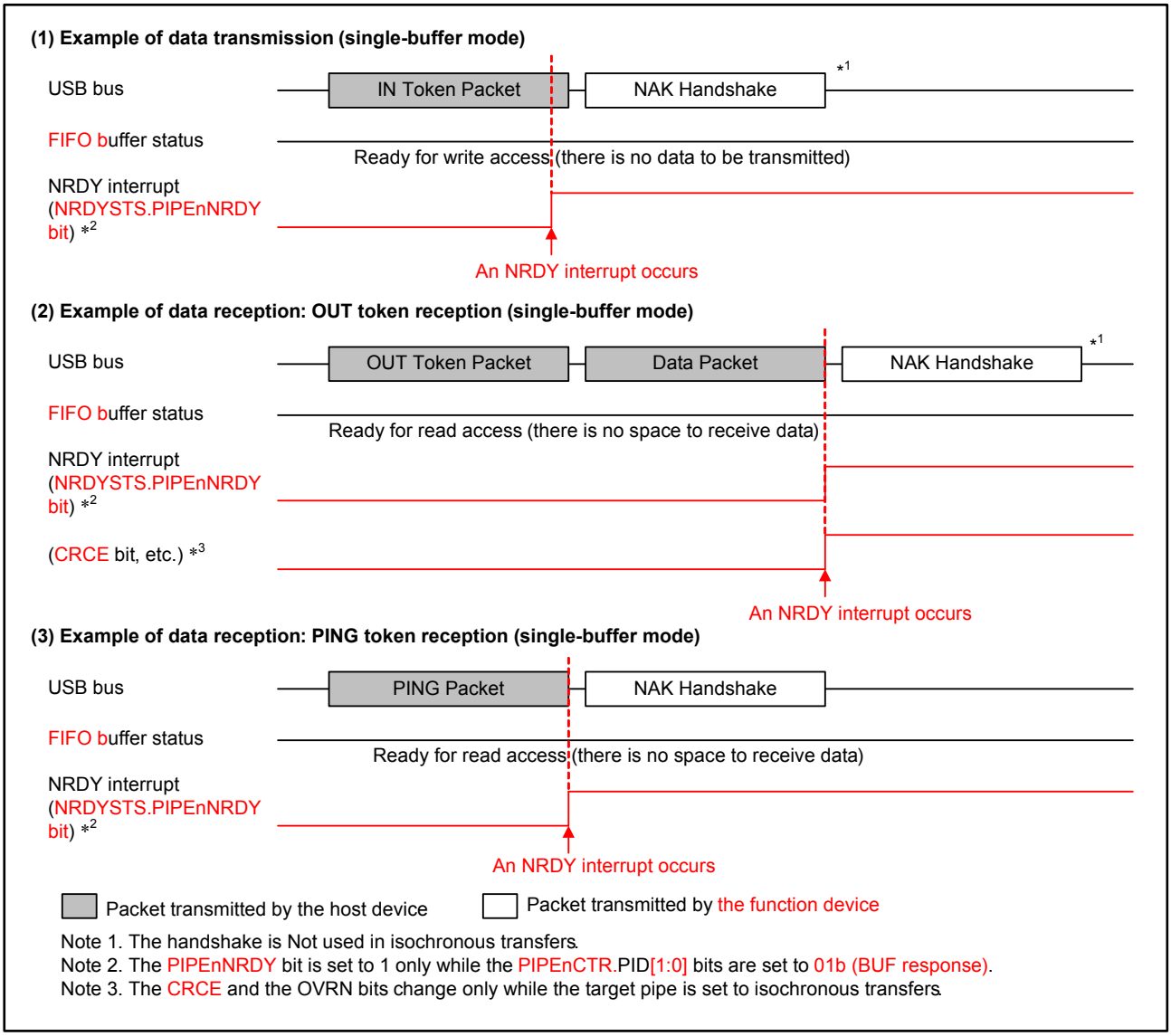


Figure 28.12 Timing of NRDY Interrupt Generation (When Function Controller Function is Selected)

•Page 1362 of 2020

Figure 28.13 is corrected as follows:

Before correction

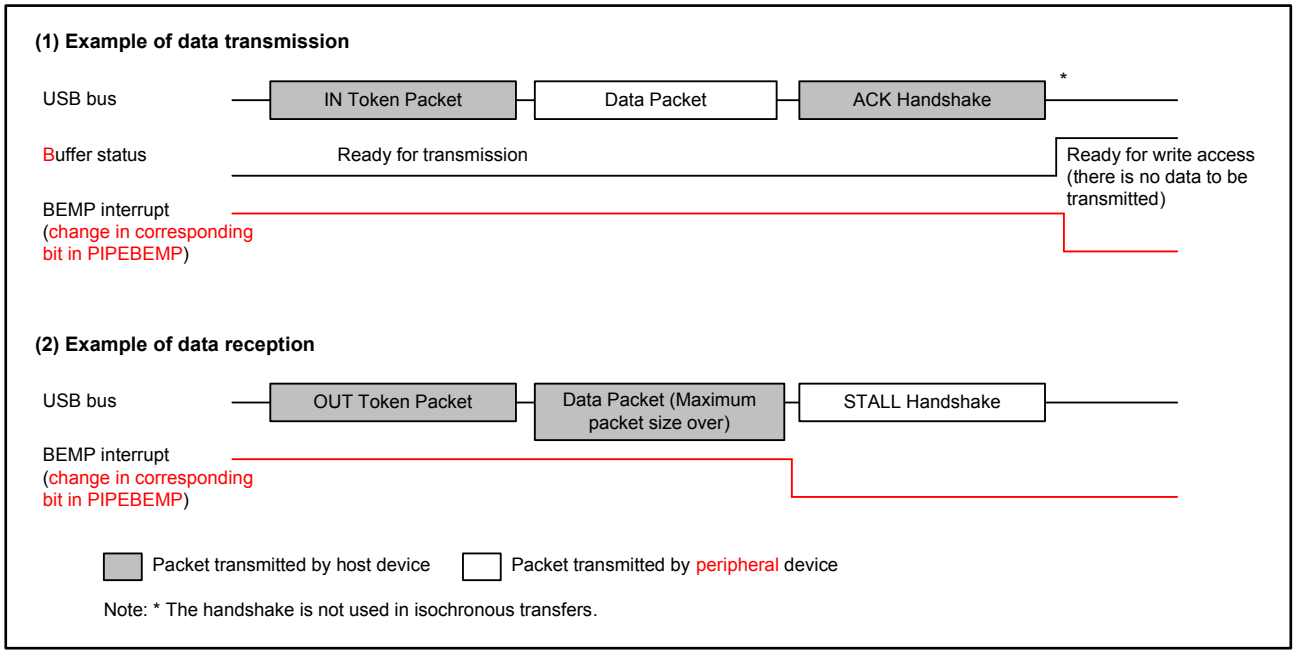


Figure 28.13 Timing of BEMP Interrupt Generation (When Function Controller Function is Selected)

After correction

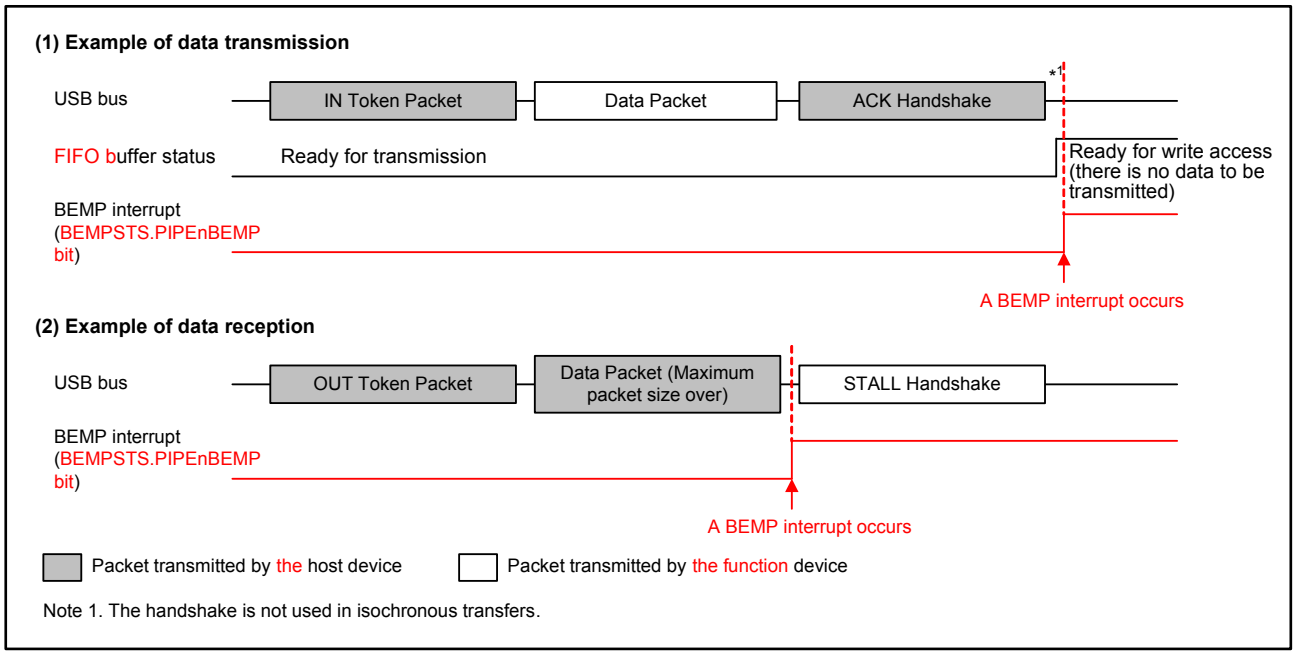


Figure 28.13 Timing of BEMP Interrupt Generation (When Function Controller Function is Selected)

•Page 1379 of 2020

The descriptions for (4) Control Transfer Auto Response Function in 28.3.6.2 are corrected as follows:

Before correction

(4) Control Transfer Auto Response Function

The USB module 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.

- Any transfer other than a control read transfer: bmRequestType 00h
- Request error : wIndex ≠ 00h
- Any transfer other than a no-data control transfer: wLength ≠ 00h
- Request error: wValue > 7Fh
- Control transfer of a device state error: DVSQ = 011b (Configured)

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

After correction

(4) Control Transfer Auto Response Function

The USB module 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 1380 of 2020

The descriptions for (1) Counter Initialization in 28.3.8.1 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 bits are initialized.
- Buffer memory initialization using the ACLRM bit:
  The IITV bits are not initialized but the count value is initialized. Setting the 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 1384 of 2020

The descriptions for (1) Counter Initialization when Function Controller Function is Selected in 28.3.9.3 are corrected as follows:

Before correction

(1) Counter Initialization when Function Controller Function is Selected

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

• Power-on Reset

The IITV bits are initialized.

• Buffer memory initialization using the ACLRM bit

The IITV bits are not initialized but the count value is initialized. Setting the ACLRM bit to 0 starts counting from the value set in the IITV bits.

After correction

(1) Counter Initialization when Function Controller Function is Selected

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 1386 of 2020

The descriptions for “When IITV = 0” in 28.3.9.3 (3) and Figure 28.18 are corrected as follows:

Before correction

• When IITV = 0: The interval counting starts at the frame following the frame in which software has set the PID bits for the selected pipe to BUF.

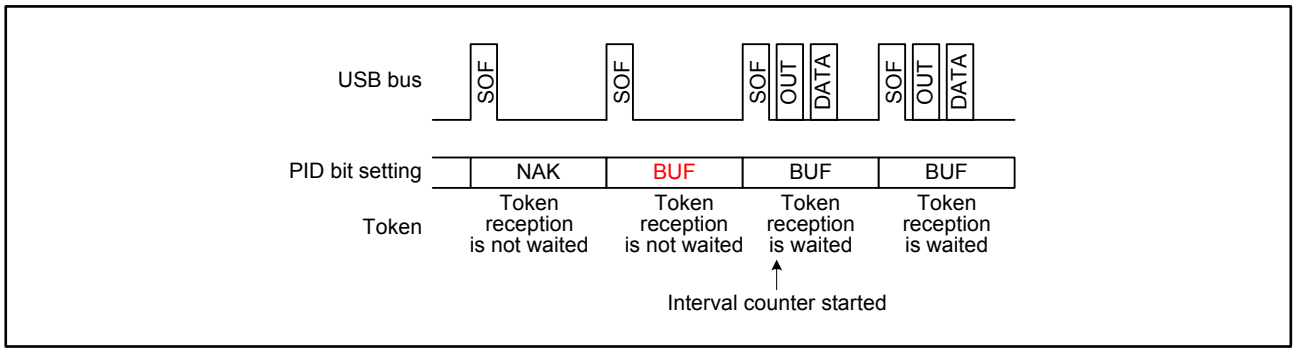


Figure 28.18 Relationship between Frames and Expected Token Reception when IITV = 0

After correction

• When IITV = 0

The interval counter starts when software has set the PID[1:0] bits for the selected pipe to BUF.

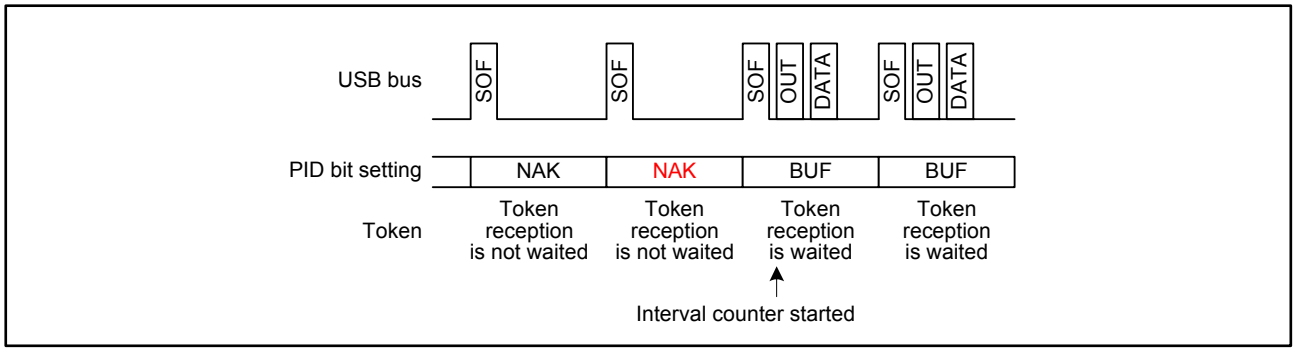


Figure 28.18 Relationship between Frames and Expected Token Reception when IITV = 0

•Page 1391 of 2020

The descriptions for SOF Interpolation Function in 28.3.10 are corrected as follows:

Before correction

23.3.10 SOF Interpolation Function

When the function controller function is selected and if data could not be received at intervals of 1 ms because an SOF packet was corrupted or missing, the USB module 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 module 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 when full-speed operation is being used, the FRNM bit in FRMNUM0 is not updated.

After correction

23.3.10 SOF Recovery Function

When the function controller function is selected and if data could not be received at intervals of 1 ms because an SOF packet was corrupted or missing, the USB module 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 module 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 when full-speed operation is being used, the FRMNUM.FRNM[10:0] bits are not updated.