32-bit RISC Microcontroller

TXZ/TXZ+ Family

Reference Manual
Full Universal Asynchronous Receiver Transmitter Circuit (FUART-B)

Revision 4.1

2023-09

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Contents

List of Figures

List of Tables

Preface

Related document

Document name
Exception
Clock Control and Operation Mode
Product Information
Input/Output Ports

Conventions

- Numeric formats follow the rules as shown below:
  Hexadecimal: 0xABC
  Decimal: 123 or 0d123 – Only when it needs to be explicitly shown that they are decimal numbers.
  Binary: 0b111 – It is possible to omit the "0b" when the number of bit can be distinctly understood from a sentence.
- "_N" is added to the end of signal names to indicate low active signals.
- It is called "assert" that a signal moves to its active level, "deassert" to its inactive level.
- When two or more signal names are referred, they are described like as [m: n].
  Example: S[3: 0] shows four signal names S3, S2, S1 and S0 together.
- The characters surrounded by ***[ ]*** defines the register.
  Example: ***[ABCD]***
- "n" substitutes suffix number of two or more same kind of registers, fields, and bit names.
  Example: ***[XYZ1], [XYZ2], [XYZ3]*** -> ***[XYZn]***
- "x" substitutes suffix number or character of units and channels in the Register List.
  In case of unit, "x" means A, B, and C ...
  Example: ***[ADACR0]***, ***[ADBCR0]***, ***[ADCCR0]*** -> ***[ADxCR0]***
  In case of channel, "x" means 0, 1, and 2 ...
  Example: ***[T32A0RUNA]***, ***[T32A1RUNA]***, ***[T32A2RUNA]*** -> ***[T32AxRUNA]***
- The bit range of a register is written like as [m: n].
  Example: Bit[3: 0] expresses the range of bit 3 to 0.
- The configuration value of a register is expressed by either the hexadecimal number or the binary number.
  Example: ***[ABCD]***<EFG> =0x01 (hexadecimal), ***[XYZn]***<VW> =1 (binary)
- Word and Byte represent the following bit length.
  Byte: 8 bits
  Half word: 16 bits
  Word: 32 bits
  Double word: 64 bits
- Properties of each bit in a register are expressed as follows:
  R: Read only
  W: Write only
  R/W: Read and Write are possible
- Unless otherwise specified, register access supports only word access.
- The register defined as reserved must not be rewritten. Moreover, do not use the read value.
- The value read from the bit having default value of "-" is unknown.
- When a register containing both of writable bits and read-only bits is written, read-only bits should be written with their default value, In the cases that default is "-", follow the definition of each register.
- Reserved bits of the Write-only register should be written with their default value.
  In the cases that default is "-", follow the definition of each register.
- Do not use read-modified-write processing to the register of a definition which is different by writing and read out.

Terms and Abbreviations

Some of abbreviations used in this document are as follows:

ACK	Acknowledgement
CTS	Clear To Send
DMA	Direct Memory Access
FIFO	First-In First-Out
FUART	Full Universal Asynchronous Receiver Transmitter
IrDA	Infrared Data Association
LSB	Least Significant Bit
MSB	Most Significant Bit
RTS	Request To Send
RZI	Return to Zero Inverted
SIR	Serial Infrared

1. Outlines

The Full Universal Asynchronous Receiver Transmitter (FUART) can operate as a transmission/reception circuit of 1 channel (FUTxTXD/FUTxRXD/FUTxCTS_N/FUTxRTS_N) per unit. And reception function using IrDA circuit. The following is a list of functions.

Function classification	Function	Operation explanation
Baud rate control	Frequency dividing of prescaler	Selectable from 1/1, 1/2, 1/4 to 1/512 of the ΦT0 frequency.
	Baud rate generator (Note1)	N + (K / 64) (N = 2 to 65535 and K = 0 to 63) dividing of the source clock frequency are possible.
Data format	Data length	Selectable 5, 6, 7 or 8 bit.
	Parity	Parity control: Enable or disable selection Parity type: Even or odd parity is selectable
	Stop bit length	Selectable 1-bit or 2-bit.
	Data transfer order	LSB first
Transmission/ reception control	FIFO function	FIFO ON and OFF is selectable.
	FIFO storage stages	Reception: 32 stages (12-bit width) Transmission: 32 stages(8-bit width)
	Error detection	Reception: Parity error, Framing error, Break error, Overrun error
	Handshake function	Transmission/reception control by handshake with FUTxCTS_N/FUTxRTS_N signal is possible.
IrDA 1.0 circuit	Data rate	Maximum 115.2 kbps (Half duplex)
	Mode	Normal IrDA mode and Low power IrDA mode
Interlocking control	Interrupt	Interrupt of combination of a transmission/reception completion interrupt, error occurrence interrupt, and a receive timeout interrupt
	DMA request (Note2)	Reception DMA request: Burst transfer or Single transfer Transmission DMA request: Burst transfer or Single transfer

Note1: Supported Maximum baud rate depend on the product. Please refer to reference manual of "Product Information" for details.

Note2: Supported DMA requests depend on the product. Please refer to reference manual of "Product Information" for details.

2. Configuration

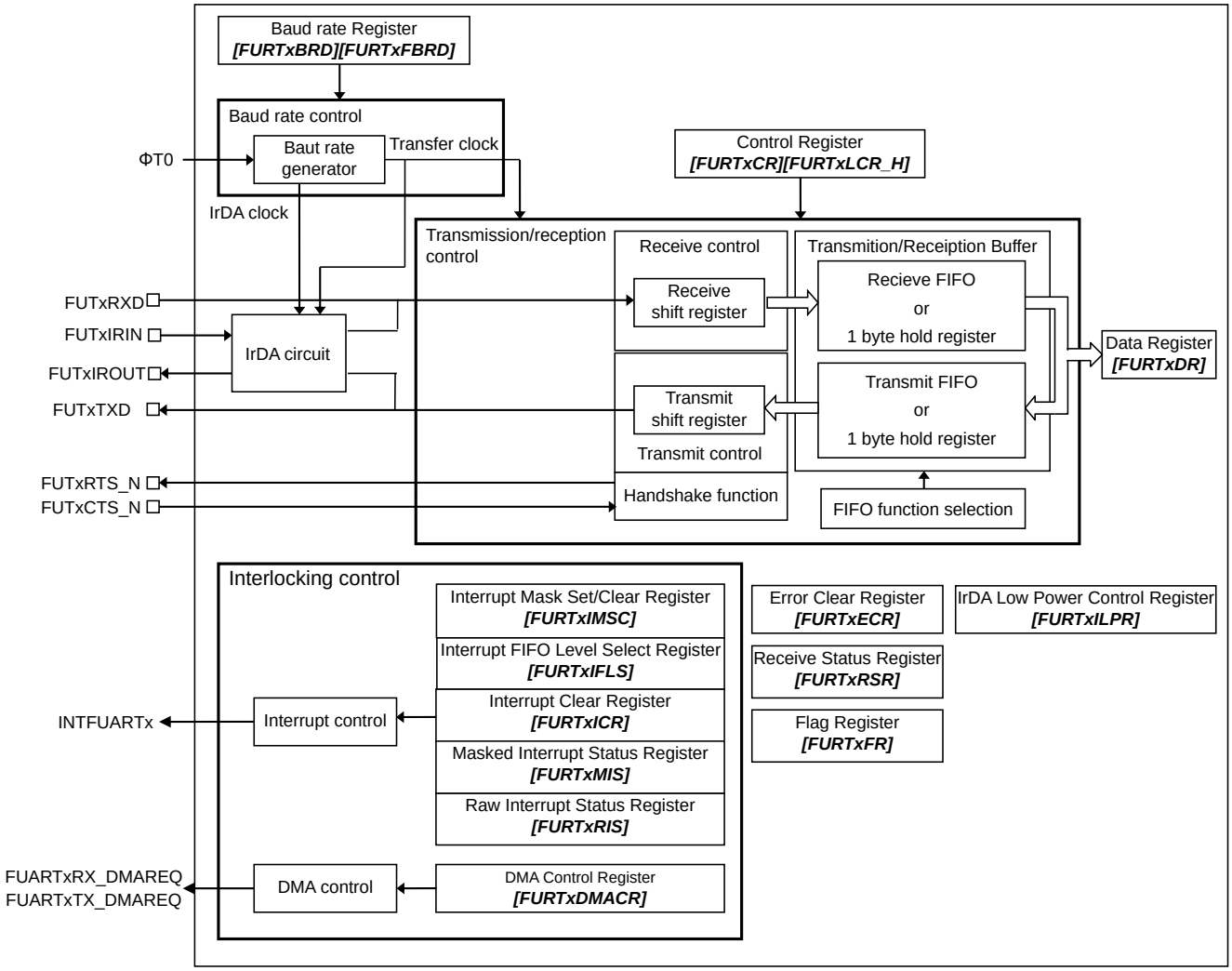

Figure 2.1 FUART block diagram

Table 2.1 List of Signals

No	Symbol	Signal name	I/O	Related reference manual
1	ΦT0	Prescaler clock	Input	Clock Selective Watchdog Timer Clock Control and Operation Mode (Note)
2	FUTxRXD	Data input pin	Input	Product Information
3	FUTxTXD	Data output pin	Output	Product Information
4	FUTxRTS_N	Request to send signal pin	Output	Product Information
5	FUTxCTS_N	Clear to send signal pin	Input	Product Information
6	FUTxIRIN	IrDA reception decoder input pin	Input	Product Information
7	FUTxIROUT	IrDA transmission encoder output pin	Output	Product Information
8	INTFUARTx	Transmission/Reception error interrupt	Output	Exception
9	FUARTxRX_DMAREQ	Reception DMA request	Output	Product Information
10	FUARTxTX_DMAREQ	Transmission DMA request	Output	Product Information

Note: Refer to " Clock Selective Watchdog Timer" for TXZ family and " Clock Control and Operation Mode" for TXZ+ family.

3. Function and Operation

3.1. Clock Supply

When you use FUART, please set an applicable clock enable bit to "1" (clock supply) in fsys supply stop register A (***[CGFSYSENA]***, ***[CGFSYSMENA]***), fsys supply stop register B (***[CGFSYSENB]***, ***[CGFSYSMENB]***), and fc supply stop registers (***[CGFCEN]***).

An applicable register and the bit position vary according to a product. Therefore, the register may not exist with the product. Please refer to "Clock Control and Operation Mode" of the reference manual for the details.

When attempting to stop supplying the clock, make sure to check whether the FUART is stopping. Note that when the MCU enters STOP mode, make sure to check whether the FUART is stopping as well.

3.2. Transfer Clock(Baud rate generator)

The following shows the diagram of the transfer clock generator.

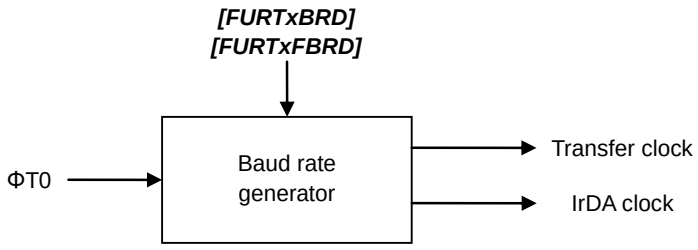

Figure 3.1 Transfer clock generator

The baud rate generator output consists of the transfer clock which controls UART transmit/receive timings and IrDA clock which controls the pulse width of IrDA encode transmission bit stream in the low power IrDA mode.

The transfer clock is generated by dividing the input clock (ΦT0) with (N + (K /64)). The division value setting is done in ***[FURTxBRD]*** (Integer baud rate register (N)) and ***[FURTxFBRD]*** (Fractional baud rate register (K)). The frequency of the transfer clock is 16 times of the baud rate.

The baud rate is calculated as follows:

$$\text{Baud rate} = \frac{\Phi T0}{\left(N + \frac{K}{64}\right)} \div 16$$

3.3. Data Format

The summary of the data formats is shown in Table 3.1 and Figure 3.2.
A data length, a data transfer order, a parity, a STOP bit length, and a data signal inversion can be selected.

Table 3.1 Transfer mode

Data length	Data transfer order	Parity	STOP bit length (transmission)
5-bit	LSB first	Presence/absence	1-bit or 2-bit
6-bit			
7-bit			
8-bit			

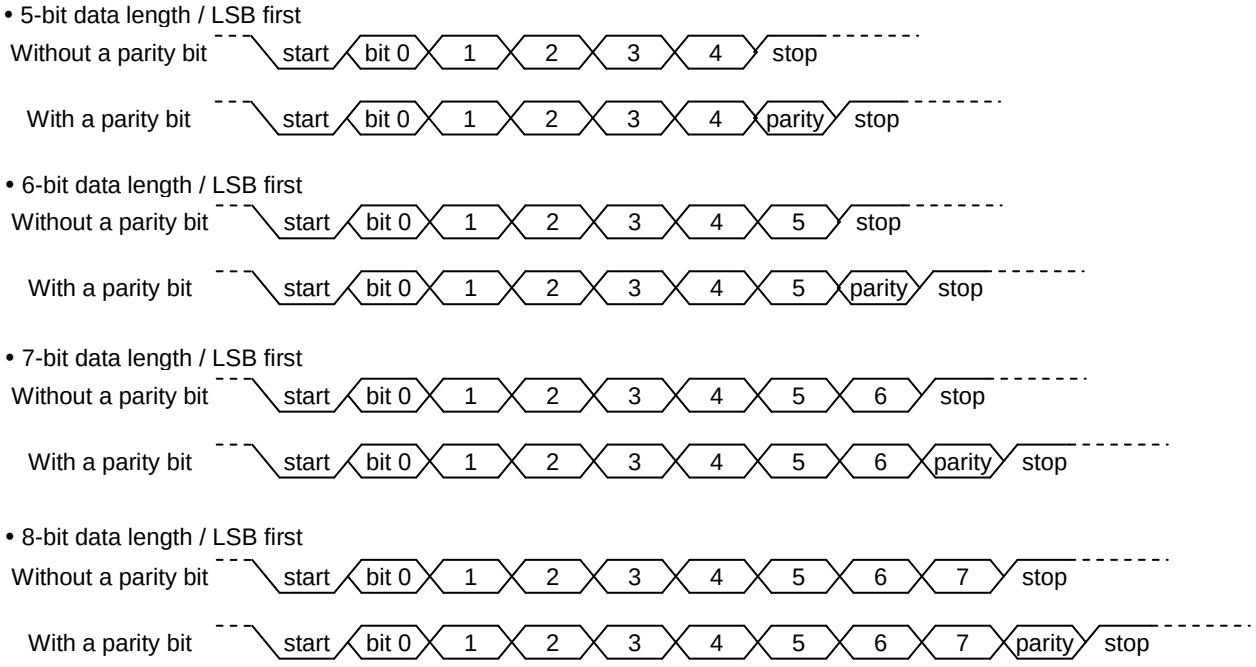

Figure 3.2 Data format

(1) Data length

Selected from among 5, 6, 7, and 8-bit.

(2) Parity control

When ***[FURTxLCR_H]***<PEN> (Parity Enable) is set to "1", the parity is enabled. The even or odd parity can be selected by ***[FURTxLCR_H]***<EPS> (Even Parity Selection).

The parity is automatically generated from the reception data at data reception. The generated parity is compared with the received parity. If they are not identical, the parity error is generated.

(3) STOP bit length

[FURTxLCR_H]<STP2> (STOP bit length selection) can set the STOP bit length in the UART transmission mode to 1-bit or 2-bit. At data reception, the STOP bit length is handled as 1 bit regardless of the setting value of this bit.

3.4. Reception Buffer / Transmission Buffer

Received data and transmitted data are stored in a reception buffer and a transmission buffer, respectively.

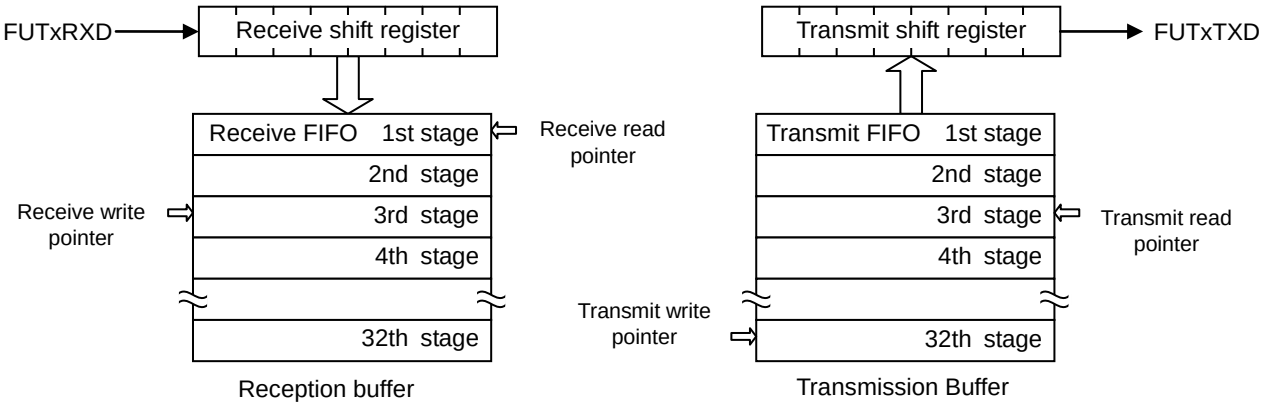

Figure 3.3 Data buffer configuration

3.4.1. Reception Buffer

Data and 4 status bits (Break error, Framing error, Parity error, and Overrun error) which are input from FUTxRXD pin are stored in the receive shift register (1 frame). Then the frame data is transferred to the receive FIFO. And the receive write pointer moves to the next stage. When the receive FIFO is read, the receive read pointer moves to the next stage. The data in the receive FIFO can be read from ***[FURTxDR]*** (Data Register).

If ***[FURTxLCR_H]***<FEN> is set to "0" (Disabled), the FIFO can be used as 1 byte hold register.

When the receive FIFO becomes full, ***[FURTxFR]***<RXFF> is set to "1". And when the receive FIFO becomes empty, ***[FURTxFR]***<RXEF> is set to "1".

The reception interrupt can be generated if the reception interrupt mask is permitted (***[FURTxIMSC]***<RXIM> = "1"). The FIFO level at which the reception interrupt is generated is selected by ***[FURTxIFLS]*** <RXIFLSEL[2:0]>.

3.4.2. Transmission Buffer

The data written to ***[FURTxDR]*** (Data Register) is stored in the transmit FIFO, and the transmit write pointer moves to the next stage. When data transmission is enabled, the data in the transmit FIFO is transferred to the transmit shift register. Then, the data is output on FUTxTXD pin. When the transmit FIFO data is transferred, the transmit read pointer moves to the next stage.

If ***[FURTxLCR_H]***<FEN> is set to "0" (Disabled), the FIFO can be used as 1 byte hold register.

When the transmit FIFO becomes full, ***[FURTxFR]***<TXFF> is set to "1".

3.5. Data Transfer

The reception or transmission data is stored in two 16-byte FIFO's. The receive FIFO has more 4-bit status per byte data.

Transmission data is written to the transmit FIFO. The transmission of a data frame starts according to the parameters set in ***[FURTxLCR_H]***. The transmission continues until the FIFO becomes empty. Once data is written to the transmit FIFO (the FIFO is not empty), ***[FURTxFR]***<BUSY> is set to "1" and it keeps "1" during the transmission. ***[FURTxFR]***<BUSY> becomes "0" when the transmit FIFO becomes empty and the last data which includes its end bit is transmitted from the shift register. ***[FURTxFR]***<BUSY> is set to "1" when the transmit FIFO is not empty regardless that UART operation is enabled or disabled.

A data is sampled by three continuous transfer clocks. If two or more sampled values are the same, the value is used as the correct data.

During the state of UART reception wait (FUTxRXD is "1" continuously), if Low data input (Start bit reception) is detected, the data is sampled at the 8th cycle of the transfer clock (the center of a bit cycle). The start bit is detected when FUTxRXD is still Low at the 8th cycle of the transfer clock.

When the start bit is detected, the data bit is sampled at the 15th cycle of the transfer clock (after 1-bit cycle) according to the data length set in ***[FURTxLCR_H]***<WLEN[1:0]>. Then the parity bit is checked if ***[FURTxLCR_H]***<PEN> is "1" (Parity is enabled).

FUTxRXD is High at the last bit, the stop bit is detected. Otherwise, the framing error occurs. After all data is received, the data and its error bits are stored to the receive FIFO.

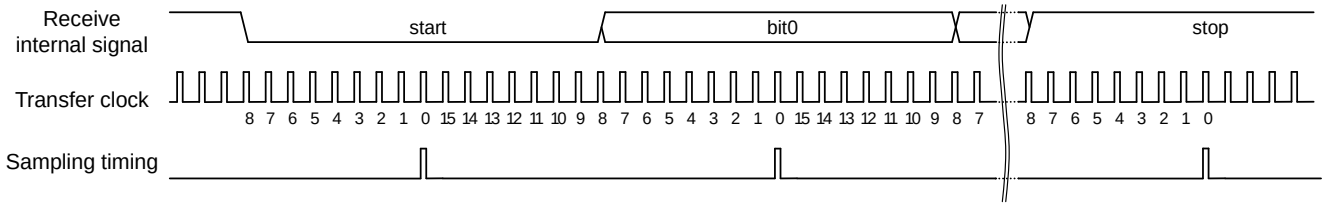

Figure 3.4 Reception timing

3.6. Data Reception

3.6.1. Basic Operation

If ***[FURTxCR]***<RXE> is set to "1", the data reception is enabled. When the START bit is detected, the data reception starts. The data bits and the STOP bit are received. When ***[FURTxCR]***<RXE>=1 is set and FUTxRXD is "Low", it is regarded immediately as START bit detection.

After one frame data reception completes, the reception data is transferred to the receive FIFO. If FIFO is disabled, the data is transferred to 1 byte hold register. The reception interrupt is generated when data are stored to a Fill level in the receive FIFO, if the reception interrupt mask is permitted (***[FURTxIMSC]***<RXIM>=1). The Fill level is set in ***[FURTxIFLS]***<RXIFLSEL[2:0]> (the reception Fill level setting).

If "0" is set to ***[FURTxCR]*** <RXE> during reception, it stops after reception of one frame being received is completed.

The followings are examples to generate the reception interrupt.

- FIFO is disabled (***[FURTxLCR_H]***<FEN>=0):
  When data is transferred from the reception shift register to 1 byte hold register.

- FIFO reception level is "1/8" (***[FURTxIFLS]***<RXIFSEL[2:0]>=000):
  FIFO level is set to 1/8 full (4 Bytes). When the fourth byte data is stored in the receive FIFO (after STOP bit is received).

The reception interrupt is cleared by the bellow procedures in above example.

- FIFO is disabled (***[FURTxLCR_H]***<FEN>=0):
  By reading from ***[FURTxDR]*** or setting ***[FURTxICR]***<RXIC> to “1”.

- FIFO reception level is "1/8" (***[FURTxIFLS]***<RXIFSEL[2:0]>=000):
  When the FIFO level is 3 bytes or less by reading from ***[FURTxDR]***, or by setting ***[FURTxICR]***<RXIC> to “1”.

3.6.2. Reception Error Handling

When an error occurs, the corresponding error bits in ***[FURTxRSR]*** and ***[FURTxDR]*** (Data Register) are set to "1". And the corresponding error interrupt is generated if its error interrupt mask is permitted (***[FURTxIMSC]***<OEIM>,<BEIM>,<PEIM>, or <FEIM>=0).

The error interrupt is cleared by setting the corresponding bit in ***[FURTxICR]*** to “1”.

The following error detections are done during the data reception.

- Parity error
  If the parity is enabled (***[FURTxLCR_H]***<PEN>=1), the parity which is generated by the reception data and the parity defined by ***[FURTxLCR_H]***<EPS> and ***[FURTxLCR_H]***<SPS> are compared after whole data is received. When they do not match, the parity error occurs.

- Framing error
  If the received STOP bit is "0", the framing error occurs.

- Break error
  When all data are "0" between the START bit and the STOP bit, the break error occurs.

- Overrun error
  All stages in the receive FIFO store data. Then, if the START bit of the next frame is detected, the overrun error occurs.
  Even when the overrun error occurs, the data reception continues. Receive operation continues and receive data is overwritten in receive shift register, but it is not written to receive FIFO. When ***[FURTxDR]***(Data Register) is read and some space is generated in the receive FIFO, the data in the reception shift register is transferred to the receive FIFO. If FIFO is disabled, the overrun error occurs when data is received before the previous data has not been read yet.

3.7. Data Transmission

3.7.1. Basic Operation

[FURTxCR]<TXE> should be set to "1" to enable the data transmission. If data exists in the transmit FIFO, the transmission starts. If data is written to the transmit FIFO, the transmission starts.

When data transmission starts, the data is transferred from the transmit FIFO to the transmission shift register. After START bit is output, data, a parity bit (if the parity is enabled), and STOP bit is transmitted. If the transmission interrupt mask is permitted (***[FURTxIMSC]***<TXIM>=1), the transmission interrupt is generated when the data level in the transmit FIFO becomes the Fill level (***[FURTxIFLS]***<TXIFLSEL[2:0]> (the transmit Fill level setting) or less.

The data transmission continues until all data in the transmit FIFO are transmitted.

The followings are examples to generate the transmission interrupt.

- FIFO is disabled (***[FURTxLCR_H]***<FEN>=0):
  When data is transferred from the transmission shift register to 1 byte hold register.

- FIFO transmission level is "1/8" (***[FURTxIFLS]***<TXIFSEL[2:0]>=000):
  FIFO level is set to 1/8-full (4 bytes). When the fifth byte data is read from the transmit FIFO (when STOP bit is transmitted) and the data count in FIFO becomes 4 bytes.

The transmission interrupt is cleared by the bellow procedures in above example..

- FIFO is disabled (***[FURTxLCR_H]***<FEN>=0):
  By writing to ***[FURTxDR]*** or setting ***[FURTxICR]***<TXIC> to “1”.

- FIFO transmission level is "1/8" (***[FURTxIFLS]***<TXIFSEL[2:0]>=000):
  When the FIFO level is 5 bytes or more by writing to ***[FURTxDR]***, or by setting ***[FURTxICR]***<TXIC> to “1”.

3.7.2. Transmission of Break Error

While <BRK> is set to "1", FUTxTXD will be continued to output "low" after transmission of current on-going frame is complete. To generate break state, <BRK> must be kept "1" for two frame period. If break state is generated, the contents of transmit FIFO is not influenced. When the break is not transmitted, <BRK> must be cleared to "0".

3.8. Interrupt

3.8.1. Interrupt Request Flag Generation Circuit

(1) Generation circuit of break, parity, and framing error flags

An interrupt request flag (***[FURTxRIS]***) is changing in real-time associated with F/F. Each flag is cleared when corresponding interrupt clear register (***[FURTxICR]***) is written.

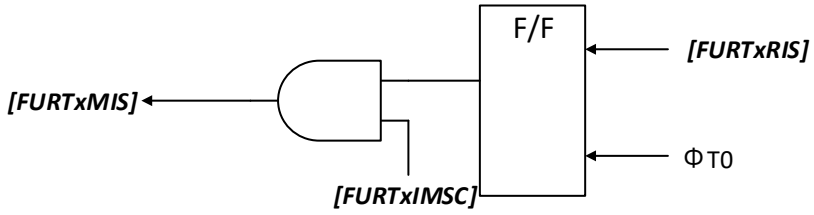

Figure 3.5 Interrupt request flag generation circuit (1)

(2) Overrun error flag generation circuit

An interrupt request flag(***[FURTxRIS]***) is changing with overrun errors in real time. The status is not maintained. An overrun flag is cleared by reading receive FIFO.

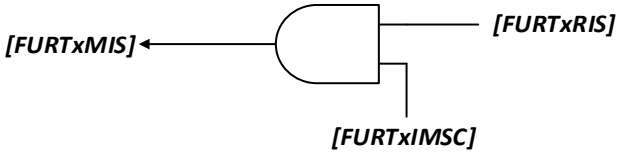

Figure 3.6 Interrupt request flag generation circuit (2)

3.8.2. FUART Interrupt

A maskable combination interrupt is output for each interrupt factor.

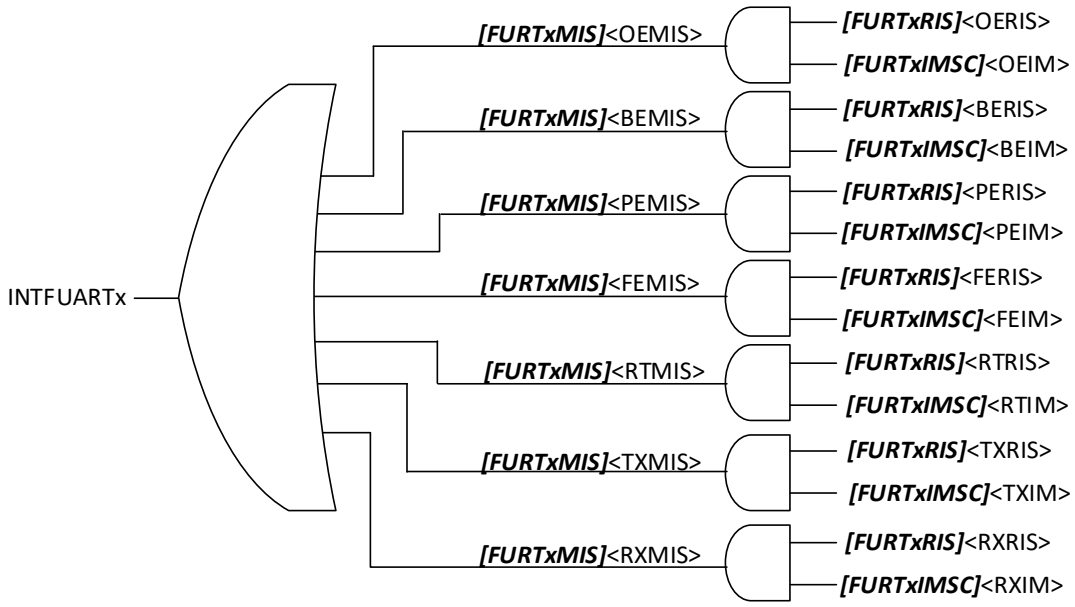

Figure 3.7 FUART combination interrupt

The timing of the interrupt generation is as shown in Table 3.2.

Table 3.2 Interrupt Generation Timing

Interrupt source	Interrupt generation timing
Overrun error generation	After a STOP bit is received when FIFO is full.
Break error interrupt	After a STOP bit is received.
Parity error generation	After a parity data is received.
Framing error generation	After Bit data that generates frame over is received.
Reception timeout interrupt	After data is received in receive FIFO, then 511 clocks of the transfer clock have elapsed.
Transmission interrupt	When 1 byte hold register is used (When FIFO is unused): The transmission interrupt occurs when a data is transferred from 1st stage of FIFO to the transmission shift register (when the transmission buffer becomes empty). (Note2)
	When FIFO is used: When the data count in FIFO becomes a set level at the start of STOP bit transmission (after MSB data is transmitted). (Note1)
Reception interrupt	When 1 byte hold register is used (When FIFO is unused): After STOP bit is received.
	When FIFO is used: After STOP bit is received when the data count in FIFO becomes a set level.

Note1: In this table, a STOP bit means the last STOP bit. (A STOP bit length is selectable with ***[FURTxLCR_H]***<STP2>.)

Note2: In the event that a transmission interrupt is cleared by software while data is being transferred (when ***[FURTxICR]***<TXIC> is set to "1"), if the timing of generating a STOP bit after the completion of transmission overlaps with writing data to the transmission buffer, the transmission interrupt does not occur.

3.9. Handshake Function

Hardware flow control is available. FUTxRTS_N pin and FUTxCTS_N pin are used to control the serial data flow.

Figure 3.8 shows the hardware flow control between two different devices.

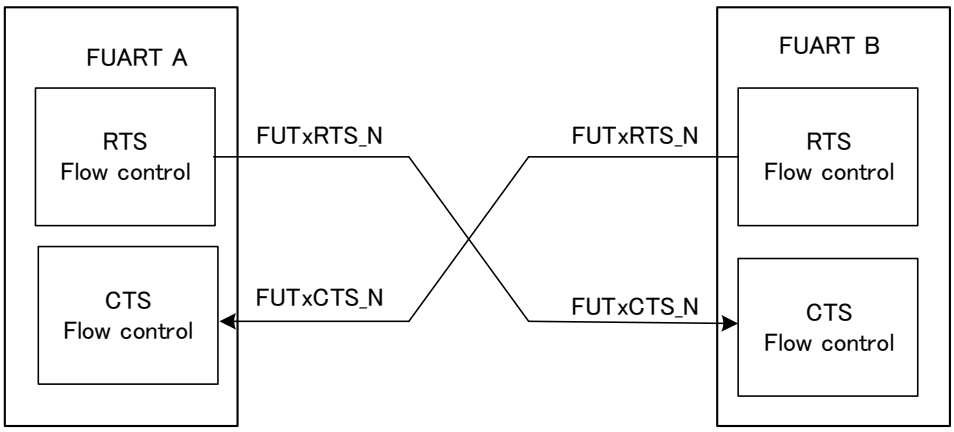

Figure 3.8 Hardware flow control

- RTS Flow control

  RTS flow control is linked to the receive FIFO level that can be set with ***[FURTxIFLS].*** While RTS flow control is enabled, FUTxRTS_N is asserted if the data count in the receive FIFO is less than the FIFO level. When the receive FIFO is the FIFO level or more, FUTxRTS_N is deasserted, which shows no more space to store reception data.

  After data is read out from the receive FIFO and it becomes less than FIFO level, FUTxRTS_N is asserted again. The receive FIFO level is set in ***[FURTxIFLS]***<RXIFLSEL[2:0]>.

  Even when RTS flow control is disabled, communication is available.

- CTS Flow control

  When CTS flow control is enabled, FUTxCTS_N is checked before data transmission. If FUTxCTS_N is asserted, the data is transmitted. If, not, the data is not transmitted.

  When FUTxCTS_N is asserted and the transmit FIFO is not empty, data is transmitted. If the transmit FIFO is empty, data is not transmitted even though FUTxCTS_N is asserted.

  While CTS flow control is enabled, and when FUTxCTS_N is deasserted, the data transmission stops after a current transmission completes.

  Even when CTS flow control is disabled, communication is available.

Table 3.3 Hardware flow control bits

[FURTxCR]		FUTxRTS_N	Description
<CTSEN>	<RTSEN>		
1	1	0(Note)	RTS and CTS flow controls are enabled.
1	0	1	Only CTS flow control is enabled.
0	1	0(Note)	Only RTS flow control is enabled.
0	0	1	Neither RTS nor CTS is enabled.

Note: While ***[FURTxCR]***<RTSEN>=1(Enable), and unless the data count in the receive FIFO reaches the FIFO level, FUTxRTS_N=0(Enable) is not set.

3.10. DMA Request

DMA request consists of a single request and a burst request. Supported DMA requests depend on the product. Please refer to reference manual of "Product Information" for details.

- Reception DMA request

  This is enabled when ***[FURTxDMACR]***<RXDMAE> is set to "1".
  When one or more data exist in FIFO, a single request is generated.
  When the data count becomes the Fill level set in ***[FURTxIFLS]***<RXIFSEL[2:0]> or more, or when the data count is more than the Fill level at DMA transfer end, a burst request is generated.

- Transmission DMA request

  This is enabled when ***[FURTxDMACR]***<TXDMAE> is set to "1".
  When FIFO has one or more empty stages, a single request is generated.
  When the data count becomes the Fill level set in ***[FURTxIFLS]***<TXIFSEL[2:0]> or less, or when the data count is less than the Fill level at DMA transfer end, a burst request is generated.

Burst requests and single requests may be worked at the same time. When the receive FIFO level is set to ***[FURTxIFLS]***<RXIFLSEL[2:0]>=000 (receive FIFO level 1/8 setting=4 bytes) and the receive FIFO contains more than 4 bytes of data. When the data in the receive FIFO becomes less than 4 bytes, only the single DMA request works.

For example, 19 bytes should be received. If the FIFO level is set to 4 bytes, DMA controller transfers 4 bytes 4 times in the burst mode and 1byte 3 times in the single mode to complete the stream.

3.11. IrDA Circuit

The IrDA circuit can transfer data compliant with IrDA1.0 standard. The maximum baud rate is 115.2 kbps (Half duplex). The IrDA circuit configuration is shown in Figure 3.9.

The IrDA circuit is enabled when ***[FURTxCR]***<UARTEN> and <SIREN> are set to "1". Data is transmitted or received through FUTxIROUT pin or FUTxIRIN pin, respectively. FUTxTXD is fixed to "High" and FUTxRXD pin is unavailable.

If ***[FURTxCR]***<SIREN> is set to "0", FUTxTXD and FUTxRXD pins are available, FUTxIROUT pin is fixed to "0" (Light pulse is not generated), and FUTxIRIN pin is unavailable.

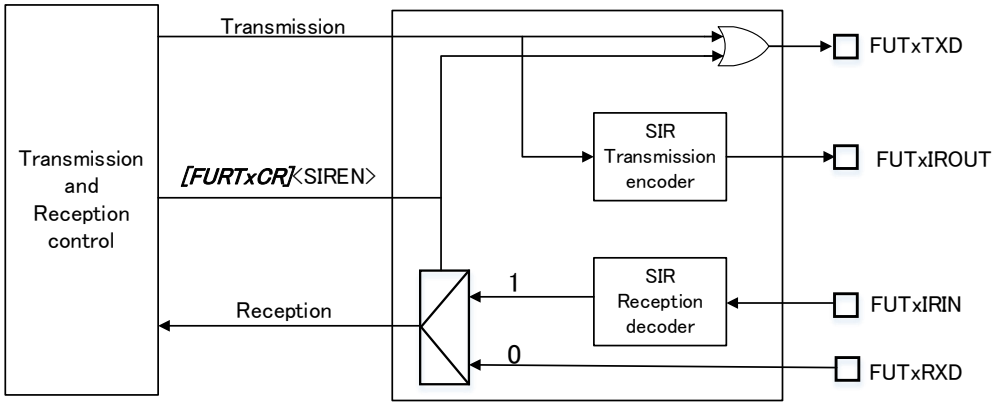

Figure 3.9 IrDA circuit

3.11.1. IrDA SIR Transmission Encoder

SIR transmission encoder modulates a transmission bit stream from the transmission and reception control circuit using RZI modulation.

In the normal IrDA mode (***[FURTxCR]***<SIRLP>=0), the transmission pulse width is generated by using 3/16 bit period. One bit period is 16 cycles of the transfer clock. And the pulse width is 3 times of the transfer clock period. Figure 3.10 shows IrDA data modulation waveform.

In the low power IrDA mode (***[FURTxCR]***<SIRLP>=1), the transmission pulse width is 3 times of IrDA clock period regardless of the bit rate. The low power IrDA mode can reduce power consumption, but the distance of communication may be shortened. IrDA clock range should be 1.42 MHz<$f_{irdaclk}$<2.12 MHz. It should be realized by setting the suitable divisor value of the IrDA low power counter to ***[FURTxILPR]***<ILPDVSR[7:0]>.

The IrDA mode can be selected by ***[FURTxCR]***<SIRLP>.

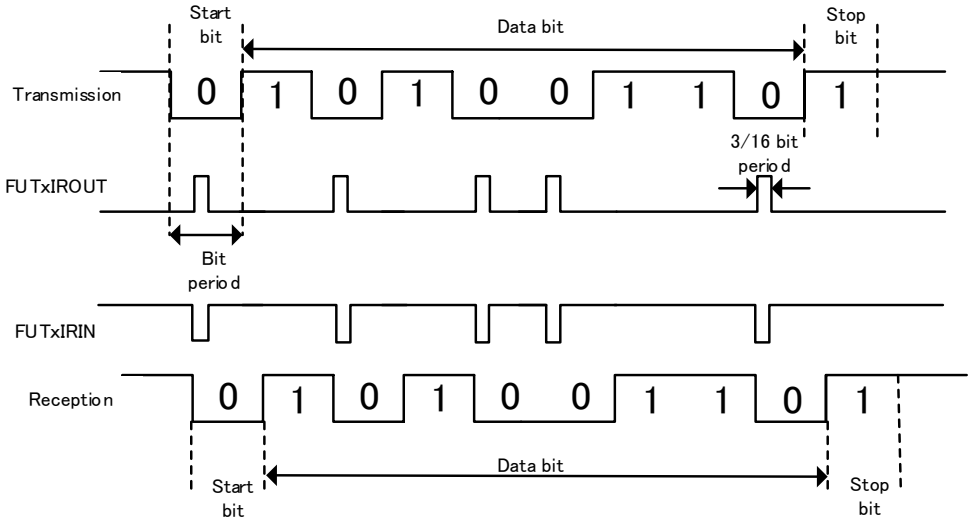

Figure 3.10 IrDA data modulation waveform

3.11.2. IrDA SIR Reception Decoder

SIR reception decoder demodulates a return-to-zero bit stream from an infrared decoder and inputs it to the transmission and reception control circuit. Normally, in the IDLE state, the input of the decoder is set to "High". The output of the transmission encoder (FUTxIROUT) has a reverse polarity of the input of the reception decoder (FUTxIRIN).

START bit is detected when the decoder input is "Low". This "Low" is detected, and the decoder input is still "Low" after 1 IrDA clock cycle elapses, then START bit is valid. This is true regardless of the normal IrDA mode or the low power IrDA mode.

4. Details of Registers

4.1. List of Registers

The control registers and their addresses are shown as follows:

Peripheral Function		Channel/Unit	Base address	
			TYPE 1	TYPE 2
Full Universal Asynchronous Receiver Transmitter	FUART	ch0	0x40048000	0x400A8000
		ch1	0x40049000	0x400A9000

Note: The channel/unit and base address type are different by products. Please refer to "Product Information" of the reference manual for the details.

Register name		Address (Base+)
Data Register	***[FURTxDR]***	0x0000
Receive Status Register	***[FURTxRSR]***	0x0004
Error Clear Register	***[FURTxECR]***	
Flag Register	***[FURTxFR]***	0x0018
IrDA Low Power Counter Register	***[FURTxILPR]***	0x0020
Integer Baud rate Register	***[FURTxBRD]***	0x0024
Fractional Baud rate Register	***[FURTxFBRD]***	0x0028
Line Control Register	***[FURTxLCR_H]***	0x002C
Control Register	***[FURTxCR]***	0x0030
Interrupt FIFO Level Select Register	***[FURTxIFLS]***	0x0034
Interrupt Mask Set/Clear Register	***[FURTxIMSC]***	0x0038
Raw Interrupt Status Register	***[FURTxRIS]***	0x003C
Masked Interrupt Status Register	***[FURTxMIS]***	0x0040
Interrupt Clear Register	***[FURTxICR]***	0x0044
DMA Control Register	***[FURTxDMACR]***	0x0048

Note: When control registers are re-set, disable FUART to operate. If the operation is disabled during receiving or transmitting, FUART will stop after on-going transmission is complete.

4.2. Details of Registers

4.2.1. *[FURTxDR]* (Data Register)

Bit	Bit Symbol	After Reset	Type	Function
31:12	-	0	R	Read as "0".
11	OE	0	R	Overrun error 0: No error 1: Error If FIFO has been full when receiving data, this bit is set to "1". When FIFO has an empty space in the FIFO and a new data can be written to FIFO, this bit is cleared to "0".
10	BE	0	R	Break error 0: No error 1: Error Break condition (FUTxRXD_N input was held "low" for longer than a full-word transmission time defined as START, data, Parity and STOP bits) is detected, this bit is set to "1". If FIFO is enabled, this error is input to the top of FIFO. If a break error occurs, "0" is stored in FIFO as data. Next data reception is enabled after FUTxRXD input is "1" (marking state) and the START bit is received.
9	PE	0	R	Parity error 0: No error 1: Error When this bit is set to "1", received data parity does not match with the parity programmed with ***[FURTxLCR_H]***<EPS> and <SPS>. If FIFO is enabled, this error is input to the top of FIFO.
8	FE	0	R	Framing error 0: No error 1: Error When this bit is set to "1", this indicates that received data does not include a valid STOP bit. (A valid STOP bit length is "1".) If FIFO is enabled, this error is input to the top of FIFO.
7:0	DATA[7:0]	0x00	R	Receive data
			W	Transmit data

Note: Error status can be identified by reading ***[FURTxRSR]*** as well.

4.2.2. *[FURTxRSR]* (Receive Status Register)

Both ***[FURTxRSR]*** and ***[FURTxECR]*** register are mapped on the same address

Bit	Bit Symbol	After Reset	Type	Function
31:4	-	0	R	Read as "0".
3	OE	0	R	Overrun error 0: No error 1: Error When data is received, if FIFO has already been full, this bit is set to "1". This bit is cleared to "0" by writing data to ***[FURTxECR]***. If FIFO is full, further data cannot be written. Thus, the content of FIFO is valid and only the content of the shift register is overwritten. CPU must be read data in order to empty FIFO.
2	BE	0	R	Break error 0: No error 1: Error If break condition (FUTxRXD input is held "low" for longer than a full-word transmission time defined as START, data, Parity, and STOP bits) is detected, this bit is set to "1". This bit is cleared to "0" by writing data to ***[FURTxECR]***. If FIFO is enabled, this error is input to the top of FIFO. If a break error occurs, "0" is input to FIFO as data. In the next data reception, FUTxRXD input is set to "1" (marking status), this bit is enabled after START bit is received.
1	PE	0	R	Parity error 0: No error 1: Error When this bit is "1", this indicates that the parity of received data does not match with the parity set in ***[FURTxLCR_H]***<EPS> and <SPS>. This bit is cleared to "0" by writing data to ***[FURTxECR]***. If FIFO is enabled, this error is input to the top of FIFO.
0	FE	0	R	Framing error 0: No error 1: Error If this bit is set to "1", this indicates that a valid STOP bit is not included in the received data. (A valid STOP bit length is "1".) This bit is cleared to "0" by writing data to ***[FURTxECR]***. If FIFO is enabled, this error is input to the top of FIFO.

Note1: Overrun error is immediately set when an error occurs.

Note2: ***[FURTxRSR]*** is updated when data is read from ***[FURTxDR]***. So received data must be read from ***[FURTxDR]*** before an error status is read from ***[FURTxRSR]***. This read sequence cannot be reversed. In addition, an error status can be read by reading ***[FURTxDR]***.

4.2.3. *[FURTxECR]* (Error Clear Register)

Both ***[FURTxRSR]*** and ***[FURTxECR]*** are mapped on the same address. These register functions are different in the read and write operations.

Bit	Bit Symbol	After Reset	Type	Function
31:4	-	0	R	Read as "0".
3	OE	0	W	When data is written to ***[FURTxECR]***, each framing, parity, break and overrun error are cleared. This clearing is executed regardless of a value of data. The address of this register is the same as those of ***[FURTxRSR]*** register.
2	BE	0	W	
1	PE	0	W	
0	FE	0	W	

4.2.4. *[FURTxFR]* (Flag Register)

Bit	Bit Symbol	After Reset	Type	Function
31:8	-	0	R	Read as "0".
7	TXFE	0	R	When ***[FURTxLCR_H]***<FEN>=1 0: Transmit FIFO is not empty. 1: Transmit FIFO is empty. When ***[FURTxLCR_H]***<FEN>=0 0: Transmit hold register is not empty. 1: Transmit hold register is empty.
6	RXFF	0	R	When ***[FURTxLCR_H]***<FEN>=1 0: Receive FIFO is not empty. 1: Receive FIFO is empty. When ***[FURTxLCR_H]***<FEN>=0 0: Receive hold register is not empty. 1: Receive hold register is empty.
5	TXFF	0	R	When ***[FURTxLCR_H]***<FEN>=1 0: Transmit FIFO is not empty. 1: Transmit FIFO is empty. When ***[FURTxLCR_H]***<FEN>=0 0: Transmit hold register is not empty. 1: Transmit hold register is empty.
4	RXFE	0	R	When ***[FURTxLCR_H]***<FEN>=1 0: Receive FIFO is not empty. 1: Receive FIFO is empty. When ***[FURTxLCR_H]***<FEN>=0 0: Receive hold register is not empty. 1: Receive hold register is empty.
3	BUSY	0	R	FUART busy 0: FUART transmission is stopping. 1: FUART transmission is performing. This bit is set to "1" when transmit FIFO becomes not empty regardless of whether FUART operation is enabled or not.
2:1	-	0	R	Read as "0".
0	CTS	0	R	FUTxCTS_N pin flag 0: FUTxCTS_N pin is "High". 1: FUTxCTS_N pin is "Low".

Note: <TXFE> does not indicate the status of shift register.

4.2.5. *[FURTxILPR]* (IrDA Low Power Counter Register)

Bit	Bit Symbol	After Reset	Type	Function
31:8	-	0	R	Read as "0".
7:0	ILPDVSR[7:0]	0x00	RW	IrDA low power counter divisor value: <ILPDVSR[7:0]> = (ΦT0 / $f_{irdaclk}$) The divisor value of IrDA clock counter should be set to realize 1.42 MHz < $f_{irdaclk}$ < 2.12 MHz. All bits are cleared to 0 at Reset.

Note 1: Set register before the ***[FURTxCR]***<SIRLP> is set to "1".
Note 2: 0x00 setting to <ILPDVSR[7:0]> is prohibited.

4.2.6. *[FURTxBRD]* (Integer Baud rate Register)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	Undefined.	R	Read as an undefined value.
15:0	BAUDDIVINT[15:0]	0x0000	RW	Integer baud rate divisor An integer part of baud rate divisor value (0x0002 to 0xFFFF).

Note1: A value written to ***[FURTxBRD]*** will not be valid until current on-going transmission or reception is completed.

Note2: A value written to ***[FURTxBRD]*** will be valid when data is written to ***[FURTxLCR_H]***.

Note3: The setting should be done before ***[FURTxCR]***<UARTEN> is set to "1".

Note4: The worst case (Data 8 bits + Parity) of the baud rate divisor value depends on the difference of the baud rate value between the transmitter and the receiver (Overall error), which is shown in the following table.

Overall error	<BAUDDIVINT[15:0]> (minimum value)
2.0 % or less	0x0002
2.8 % or less	0x0003
3.3 % or less	0x0004

4.2.7. *[FURTxFBRD]* (Fractional Baud rate Register)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	Undefined.	R	Read as an undefined value.
5:0	BAUDDIVFRAC[5:0]	000000	RW	Fractional baud rate divisor A fractional part of baud rate divisor value.

Note1: A value written to ***[FURTxFBRD]*** will not be valid until current on-going transmission or reception is completed.

Note2: A value written to ***[FURTxFBRD]*** will be valid when data is written to ***[FURTxLCR_H]***.

Note3: Set <BAUDDIVFRAC[5:0]> before "1" is set to ***[FURTxCR]***<UARTEN>.

Note4: The maximum value of the baud rate divisor is "65535". Therefore, the fractional part of the baud rate divisor must be set to "0", when the integer part of the baud rate divisor is "65535".

4.2.8. *[FURTxLCR_H]* (Line Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:8	-	0	R	Read as "0".
7	SPS	0	RW	Stick parity selection 0: Stick parity disabling 1: When <EPS>=0 is set, "1" is transmitted/received as parity bit. When <EPS>=1 is set, "0" is transmitted/received as parity bit. <SPS> has no meaning when <PEN> is set to "0" and the parity check and generation are disabled. For details of the table of truth value of <SPS>, <EPS> and <PEN>, refer to Table 4.1.
6:5	WLEN[1:0]	00	RW	Word length 00: 5bit 01: 6bit 10: 7bit 11: 8bit These bits indicate the number of data bits transmitted/received in the frame.
4	FEN	0	RW	Selection of FIFO permission 0: FIFO is disabled (FIFO becomes a 1byte hold register.) 1: FIFO is enabled
3	STP2	0	RW	Selection of transmission STOP bit length 0: 1bit 1: 2bit When receiving, STOP bit of 2bit length is not checked.
2	EPS	0	RW	Even parity selection 0: Odd parity 1: Even parity Control to selection of a parity bit when transmitting/receiving. If <PEN> is set to "0", when parity check and generation are disabled, this bit has no meaning.
1	PEN	0	RW	Parity enable 0: Disabled (The parity is disabled. The parity bit is not added.) 1: Enabled (Parity check and generation are enabled.)
0	BRK	0	RW	Break transmission selection 0: No break transmission 1: Performs break transmission While <BRK> is set to "1", FUTxTXD will be continued to output "low" after transmission of current on-going frame is complete. To generate break state, <BRK> must be kept "1" for two frame period. If break state is generated, the contents of transmit FIFO is not influenced. When the break is not transmitted, <BRK> must be cleared to "0".

Note: When the contents of ***[FURTxBRD]*** or ***[FURTxFBRD]*** is updated, always must write ***[FURTxLCR_H]*** in the end.

Table 4.1 List of truth value *[FURTxLCR_H]*<SPS>,<EPS> and <PEN>

Parity enable <PEN>	Even parity <EPS>	Stick parity selection <SPS>	Parity selection (Transmission or checking)
0	-	-	No transmission and check
1	1	0	Even parity transmit/ even parity receive
1	0	0	Odd parity transmit / even parity receive
1	0	1	Transmit /receive "1" as parity bit
1	1	1	Transmit/receive "0" as parity bit

4.2.9. *[FURTxCR]* (Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as "0".
15	CTSEN	0	RW	CTS hardware flow control enable 0: Disabled 1: Enabled When <CTSEN> is set to "1", CTS hardware flow control is enabled. Data is sent only when FUTxCTS_N pin is "1".
14	RTSEN	0	RW	RTS hardware flow control enable 0: Disabled 1: Enabled When <RTSEN> is set to "1", RTS hardware flow control is enabled. Data is required when receive FIFO is empty.
13:12	-	0	R	Read as "0".
11:10	-	00	RW	Write as "00".
9	RXE	0	RW	Reception enable setting 0: Disabled 1: Enabled When <RXE> is set to "1", reception is enabled. According to a value of <SIREN>, data is received using FUART function or SIR function. If the reception is disabled during receiving, reception stops after current on-going data reception is complete.
8	TXE	0	RW	Transmission enable setting 0: Disabled 1: Enabled When <TXE> is set to "1", transmission is enabled. According to a value of <SIREN>, data is received using FUART function or IrDA SIR function. If the transmission is disabled during transmitting, transmission stops after current on-going transmission is complete.
7	-	0	RW	Write as "0".
6:3	-	0	R	Read as an undefined value.
2	SIRLP	0	RW	IrDA mode 0: Normal IrDA mode 1: Low power IrDA mode
1	SIREN	0	RW	IrDA enable setting 0: Disabled 1: Enabled
0	UARTEN	0	RW	FUART enable setting 0: Disabled 1: Enabled When <UARTEN> is set to "0", FUART is disabled. If FUART is disabled during transmission or reception, FUART stops after current on-going data transmission or reception is complete. When <UARTEN> is set to "1", data is sent or received.

4.2.10. *[FURTxIFLS]* (Interrupt FIFO Level Select Register)

Bit	Bit Symbol	After Reset	Type	Function
31:6	-	0	R	Read as "0".
5:3	RXIFLSEL[2:0]	000	RW	Reception interrupt FIFO level selection 000: Receive FIFO ≥ 1/8 full 001: Receive FIFO ≥ 1/4 full 010: Receive FIFO ≥ 1/2 full 011: Receive FIFO ≥ 3/4 full 100: Receive FIFO ≥ 7/8 full Other than the above: Reserved
2:0	TXIFLSEL[2:0]	000	RW	Transmission interrupt FIFO level selection 000: Transmit FIFO ≤ 1/8 full 001: Transmit FIFO ≤ 1/4 full 010: Transmit FIFO ≤ 1/2 full 011: Transmit FIFO ≤ 3/4 full 100: Transmit FIFO ≤ 7/8 full Other than the above: Reserved

4.2.11. *[FURTxIMSC]* (Interrupt Mask Set/Clear Register)

Bit	Bit Symbol	After Reset	Type	Function
31:11	-	0	R	Read as "0".
10	OEIM	0	R/W	Overrun error interrupt mask 0: Enabled mask 1: Disabled mask
9	BEIM	0	R/W	Break error interrupt mask 0: Enabled mask 1: Disabled mask
8	PEIM	0	R/W	Parity error interrupt mask 0: Enabled mask 1: Disabled mask
7	FEIM	0	R/W	Framing error interrupt mask 0: Enabled mask 1: Disabled mask
6	RTIM	0	R/W	Receive timeout interrupt mask 0: Enabled mask 1: Disabled mask
5	TXIM	0	R/W	Transmit interrupt mask 0: Enabled mask 1: Disabled mask
4	RXIM	0	R/W	Receive interrupt mask 0: Enabled mask 1: Disabled mask
3:0	-	0	RW	Write as "0".

4.2.12. *[FURTxRIS]* (Raw Interrupt Status Register)

Bit	Bit Symbol	After Reset	Type	Function
31:11	-	0	R	Read as "0".
10	OERIS	0	R	Overrun error interrupt status 0: No interrupt request 1: Interrupt request.
9	BERIS	0	R	Break error interrupt status 0: No interrupt request 1: Interrupt request
8	PERIS	0	R	Parity error interrupt status 0: No interrupt request 1: Interrupt request
7	FERIS	0	R	Framing error interrupt status 0: No interrupt request 1: Interrupt request
6	RTRIS	0	R	Receive timeout interrupt status 0: No interrupt request 1: Interrupt request
5	TXRIS	0	R	Transmit interrupt status 0: No interrupt request 1: Interrupt request
4	RXRIS	0	R	Receive interrupt status 0: No interrupt request 1: Interrupt request
3:0	-	0	R	Read as "0".

4.2.13. *[FURTxMIS]* (Masked Interrupt Status Register)

Bit	Bit Symbol	After Reset	Type	Function
31:11	-	0	R	Read as "0".
10	OEMIS	0	R	Overrun error mask interrupt status 0: No interrupt request 1: Interrupt request
9	BEMIS	0	R	Break error mask interrupt status 0: No interrupt request 1: Interrupt request
8	PEMIS	0	R	Parity error mask interrupt status 0: No interrupt request 1: Interrupt request
7	FEMIS	0	R	Framing error mask interrupt status 0: No interrupt request 1: Interrupt request
6	RTMIS	0	R	Receive timeout mask interrupt status 0: No interrupt request 1: Interrupt request
5	TXMIS	0	R	Transmit mask interrupt status 0: No interrupt request 1: Interrupt request
4	RXMIS	0	R	Receive mask interrupt status 0: No interrupt request 1: Interrupt request
3:0	-	0	R	Read as "0".

4.2.14. *[FURTxICR]* (Interrupt Clear Register)

Bit	Bit Symbol	After Reset	Type	Function
31:11	-	0	W	Write as "0".
10	OEIC	0	W	Overrun error interrupt clear 0: Invalid 1: Clear
9	BEIC	0	W	Break error interrupt clear 0: Invalid 1: Clear
8	PEIC	0	W	Parity error interrupt clear 0: Invalid 1: Clear
7	FEIC	0	W	Framing error interrupt clear 0: Invalid 1: Clear
6	RTIC	0	W	Receive timeout interrupt clear 0: Invalid 1: Clear
5	TXIC	0	W	Transmit interrupt clear 0: Invalid 1: Clear
4	RXIC	0	W	Receive interrupt clear 0: Invalid 1: Clear
3:0		0	W	Write as "0"

4.2.15. *[FURTxDMACR]* (DMA Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:3	-	0	R	Read as "0".
2	DMAONERR	0	RW	DMA ON error 0: With control 1: Without control While this bit is set to "1", if an error occurs during receiving data, a DMA receive request, FUARTxRX_DMAREQ is disabled.
1	TXDMAE	0	RW	Transmit DMA enable selection 0: Disabled 1: Enabled
0	RXDMAE	0	RW	Receive DMA enable selection 0: Disabled 1: Enabled

Note: If transmit/receive FIFO data is transferred using DMAC, set the bus width to 8-bit.

5. Programming

5.1. Baud Rate Setting Value

As a reference, an example of a baud rate setting value is shown using a typical clock (ΦT0). The error of ΦT0 is not included. ***[FURTxBRD]***(N) and ***[FURTxFBRD]***(K) are calculated by the following formula.

- Setting baud rate and Clock condition

  Setting baud rate (bps): 115200 bps
  ΦT0=10 MHz (8 division of fsys = 80 MHz)

- Calculation formula

$$\text{Baud rate} = \frac{\Phi T0}{\left(N + \frac{K}{64}\right)} \div 16$$

1) Baud rate divisor = $(10 \times 10^6) / (16 \times 115200) = 5.4253452$
   Integer (N) = 0x05 (5) and Decimal = 0.4253452
2) Using Decimal in 1), ***[FURTxFBRD]*** value is calculated.
   $((0.4253452 \times 64)+0.5) = 27.7220928$ ➔ K = 0x1B(27)
3) Generated baud rate divisor = 5 (N) + 27 (K)/64 = 5.421875
4) Generated baud rate = (10×10^6)[ΦT0] / (16 x 5.421875) = 115273.776
5) Error=(115273.776 - 115200) / 115200 x 100 = 0.064 %

Table 5.1 Baud rate setting example at ΦT0=10 MHz

Baud rate (bps)	*[FURTxBRD]* (N) value	*[FURTxFBRD]* (K) value	Calculated value (bps)	Error (%)
9600	0x0041	0x07	9599.232	-0.008
19200	0x0020	0x23	19203.072	0.016
38400	0x0010	0x12	38387.716	-0.032
57600	0x000A	0x36	57636.888	0.064
115.2k	0x0005	0x1B	115273.775	0.064
128k	0x0004	0x39	127795.527	-0.160
256k	0x0002	0x28	256410.256	0.160

Table 5.2 Baud rate setting example at ΦT0=80 MHz

Baud rate (bps)	*[FURTxBRD]* (N) value	*[FURTxFBRD]* (K) value	Calculated value (bps)	Error (%)
9600	0x0208	0x35	9600.096	0.001
19200	0x0104	0x1B	19199.616	-0.002
38400	0x0082	0x0D	38401.536	0.004
57600	0x0056	0x34	57595.392	-0.008
115.2k	0x002B	0x1A	115190.785	-0.008
128k	0x0027	0x35	128000.000	0.000
256k	0x0013	0x22	256000.000	0.000
512k	0x0009	0x31	512000.000	0.000
2M	0x0002	0x20	2000000.000	0.000

6. Precautions

- FUTxRTS_N pin or FUTxCTS_N pin may not be assigned in some products. In such products, the corresponding functions should not be used.

7. Revision History

Table 7.1 Revision History

Revision	Date	Description
1.0	2018-01-09	First release
2.0	2018-03-02	-1. Outline Added maximum baud rate and Note1 in the table. Changed to "Note2" from "Note", due to added "Note1". -4.2.11 ***[FURTxIMSC]*** Changed the bit 4 to 10 of "Type" from "R" to "R/W". -5.1 Baud Rate Setting Value Deleted example of 1Mbps baud rate from Table 5.1. Added example of 256K/512Kbps baud rate in Table 5.1. Deleted example of 1M/2Mbps baud rate from Table 5.2. Added example of 256K/512Kbps baud rate in Table 5.2.
3.0	2018-05-28	- 1. Outline Deleted Baud rate line. Added Note1) in Baud rate generator. Modified explanation of Note1) - 5.1 Baud Rate Setting Value Deleted example of 62500 and 512Kbps from Table 5.1 Deleted example of 62500 from Table 5.2 Added example of 2M in Table 5.2 Changed Baud rate of Table 5.1 and Table 5.2 from "K" to "k"
3.1	2021-08-04	- Added "Error correction for technical datasheet of Universal Asynchronous Receiver-Transmitter".
4.0	2023-06-12	- Support TXZ+ family - “Error correction for technical datasheet of Universal Asynchronous Receiver-Transmitter“ is deleted. And its contents are added to table 3.2. - Table 3.2 Interrupt Generation Timing The description of “When 1 byte hold register is used (When FIFO is unused):” cell in transmission interrupt is changed. Note2 is added. - 3.4.1. Reception Buffer Changed ***[FURTxIMSC]***<RXIM> =0 to 1. - 3.4.2. Transmission Buffer Removed some descriptions. - 3.6.1. Basic Operation Changed ***[FURTxIMSC]***<RXIM> =0 to 1. - 3.7.1. Basic Operation Changed ***[FURTxIMSC]***<TXIM> =0 to 1. - 3.7.2 Transmission of Break Error Changed description. - 4.2.11. ***[FURTxIMSC]*** (Interrupt Mask Set/Clear Register) Changed bit4 to 10 0: Enabled mask, 1: Disabled mask
4.1	2023-09-15	- Table 2.1 List of Signals Corrected error and added Note. - 3.6.1. Basic Operation Added description. - 3.6.2. Reception Error Handling Added description. - 3.7.1 Basic Operation Added description.

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as "TOSHIBA".
Hardware, software and systems described in this document are collectively referred to as "Product".

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, lifesaving and/or life supporting medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, and devices related to power plant. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**