UART IP

DESIGN SPECIFICATION

Version: 2.0

Revision History

Date	Rev.	Author	Release note
08/11/2021	1.0	Huy Nguyen	Initial version
15/10/2024	2.0	Huy Nguyen	Support AHB lite protocol Format updating Access reserved region is cause of return HRESP error Update register attribute Implement parity check on UART receive Update sequence in Chapter 3. Update situation for received FIFO error handling Add attribute R/W1C

Table of Contents

List of Tables

List of Figures

Chapter 1 – Introduce

1.1 Description

The Universal Asynchronous Receiver/Transmitter (UART) perform serial-to-parallel conversions on data received from a peripheral device and parallel-to-serial conversion on data received from the CPU. The UART includes control timing and interrupt system that can be minimize software.

The UART includes a programmable baud generator capable of dividing the UART input clock by divisors from 1 to 65535 and producing a 16x reference clock or a 13x reference clock for the internal transmitter and received logic.

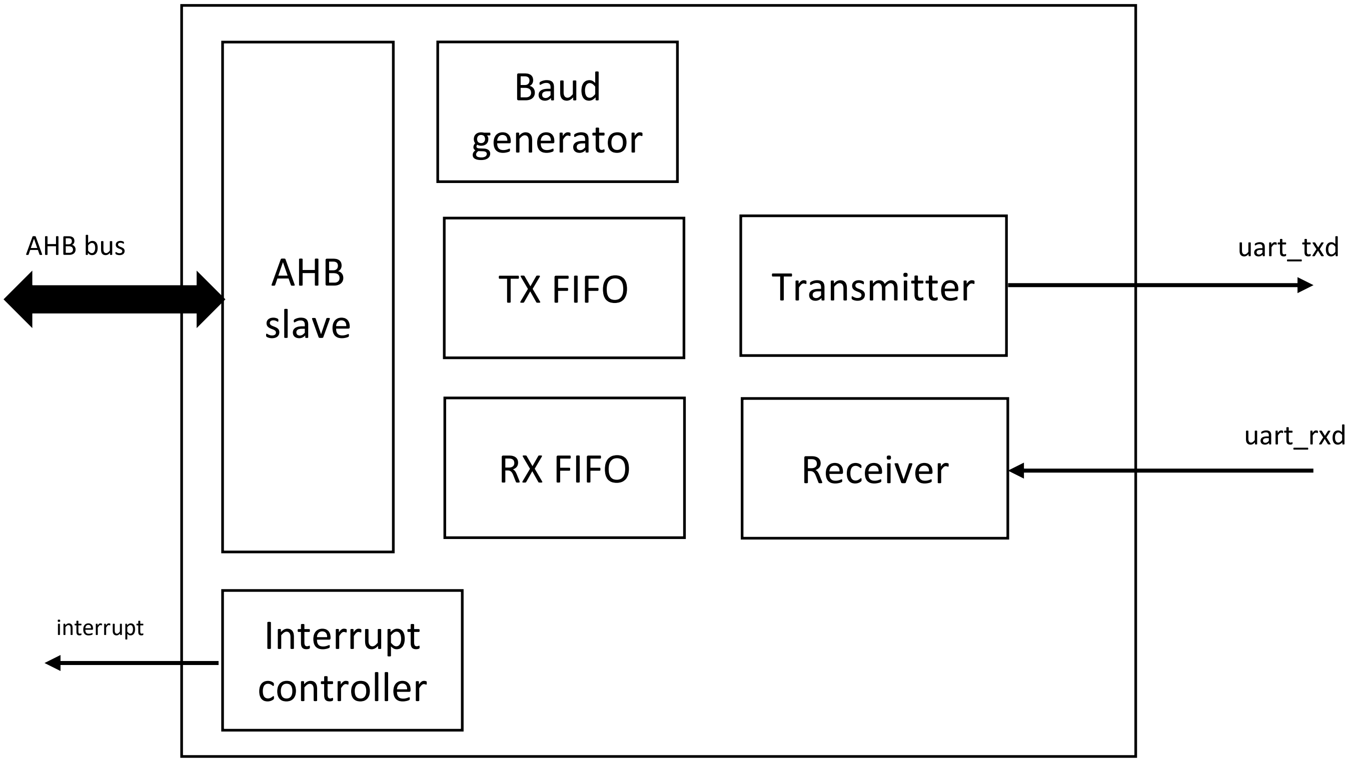
1.2 Feature

- Support AHB lite protocol
  - Support SINGLE burst transfer.
  - Support transfer size: word.
  - Not support Locked transfer, Protection control.
- Programmable baud generator.
- Support hard interrupt.
- Support separate FIFO: 16-byte TX FIFO and 16-byte RX FIFO
- Fully programmable serial-interface characteristics:
  - 5, 6, 7 and 8 bit characters.
  - Even, odd, or no-parity bit generation.
  - 1, or 2 stop bit generation

1.3 Block Diagram

A functional block diagram of the UART is shown in Figure 1-1.

Figure 1-1 Block diagram



1.4 Signal Definition

1.4.1 AHB interface signal

Table 1-1. Description of AHB inteface

Signal name	Width	I/O Type	Description
hclk	1	Input	
hresetn	1	Input	
haddr	[9:0]	Input	
hburst	[2:0]	Input	
hmastlock	1	Input	
hprot	[3:0]	Input	
hsize	[2:0]	Input	
hsel	1	Input	
htrans	[1:0]	Input	
hwdata	[31:0]	Input	
hwrite	1	Input	
hrdata	[31:0]	Output	
hreadyout	1	Output	
hresp	1	Output	

1.4.2 UART interface signal

Table 1-2. Description of UART interface

Signal name	Width	I/O Type	Description
uart_txd	1	Output	
uart_rxd	1	Input	

1.4.3 Interrupt signal

Table 1-3. Description of Interrupt

Signal name	Width	I/O Type	Description
interrupt	1	Output	

1.5 Clock Generation and Control

The processor clock generator receives a signal from an external clock source and produces a UART input clock with a programmed frequency.

The UART contains a programmable baud generator that takes an input clock and divides it by a divisor in the range between 1 and (2^{16} - 1) to produce a baud clock (BCLK).

The frequency of BCLK is sixteen times (16×) the baud rate (each received or transmitted bit lasts 16 BCLK cycles) or thirteen times (13×) the baud rate (each received or transmitted bit lasts 13 BCLK cycles). When the UART is receiving, the bit is sampled in the 8th BCLK cycle for 16× over sampling mode and on the 6th BCLK cycle for 13× oversampling mode.

The 16× or 13× reference clock is selected by configuring the OSM_SEL bit in the mode definition register (MDR). The formula to calculate the divisor is:

$$Divisor = \frac{\text{UART input clock Frequency (Hz)}}{\text{Desired Baud Rate x 16}} \qquad [MDR.OSM\_SEL = 0]$$

$$Divisor = \frac{\text{UART input clock Frequency (Hz)}}{\text{Desired Baud Rate x 13}} \qquad [MDR.OSM\_SEL = 1]$$

These divisor latches must be loaded during initialization of the UART to ensure desired operation of the baud generator.

Table 1-4. Baud rate Example for 100 –MHz UART input clock and 16x Oversampling Mode

Baud rate	Bit duration	Divisor value
2400	416.667 µs	2604
4800	208.333 µs	1302
9600	104.167 µs	651
19200	52.083 µs	325
38400	26.042 µs	163
76800	13.021 µs	81
115200	8.681 µs	54

Table 1-5. Baud rate Example for 100 –MHz UART input clock and 13x Oversampling Mode

Baud rate	Bit duration	Divisor value
2400	416.667 µs	3205
4800	208.333 µs	1602
9600	104.167 µs	801
19200	52.083 µs	401
38400	26.042 µs	200
76800	13.021 µs	100
115200	8.681 µs	67

1.6 Protocol Description

1.6.1 AHB protocol

This IP support AMBA 3 AHB lite Protocol v1.0

- Support SINGLE burst transfer, it is mean Master can perform single transfer using either:
  - SINGLE burst (HBURST[2:0] = 3'b000)
  - Undefined length burst that has a burst of length one (HBURST[2:0] = 3'b001)
- Support transfer size: word (HSIZE[2:0] = 3'b010)
- Not support Locked transfer, Protection control.

1.6.2 UART protocol

1.6.2.1 Transmission

Based on the settings chosen in the LCR register, the UART transmitter sends the following to the receiving device:

- 1 START bit
- 5,6,7 or 8 data bits
- 1 PARITY bit (optional)
- 1, 2 STOP bits

1.6.2.2 Reception

Based on the settings chosen in LCR register, the UART receiver accepts the following from the transmitting device:

- 1 START bit
- 5,6,7 or 8 data bits
- 1 PARITY bit (optional)
- 1, 2 STOP bits

1.6.2.3 Data format

The UART transmits in the following format:

1 START bit + data bits (5, 6, 7, 8) + 1 PARITY bit (optional) + STOP bit (1, 2)

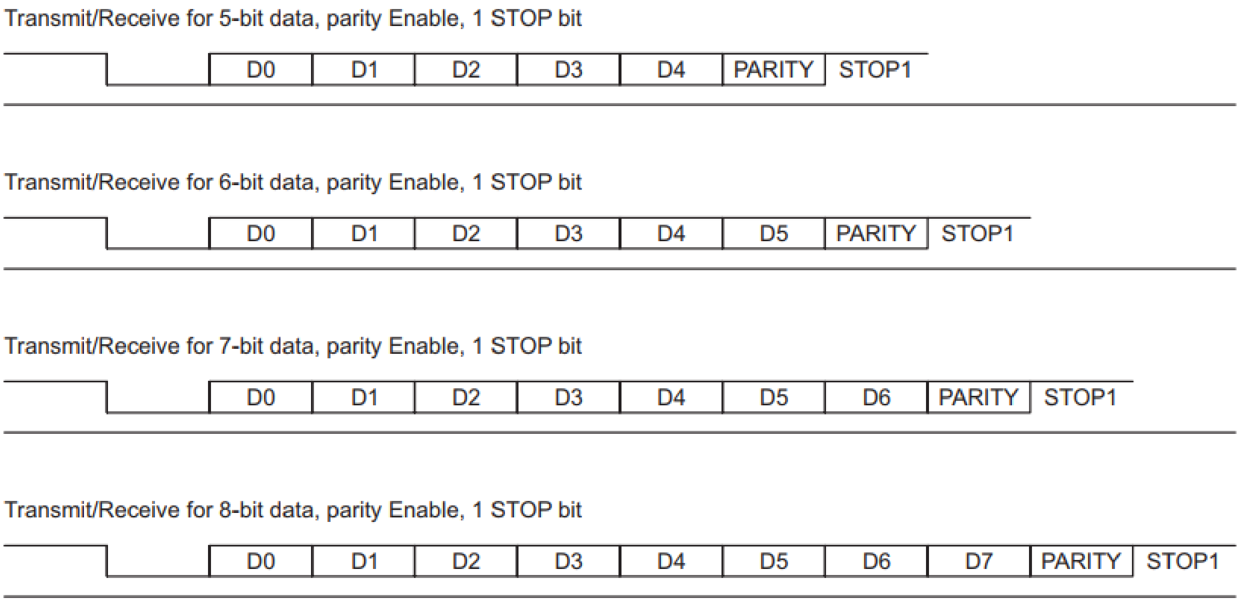
It transmits 1 START bit; 5, 6, 7, or 8 data bits, depending on the data width selection; 1 PARITY bit, if parity is selected; and 1, or 2 STOP bits, depending on the STOP bit selection.

The UART receives in the following format:

1 START bit + data bits (5, 6, 7, 8) + 1 PARITY bit (optional) + STOP bit (1, 2)

It receives 1 START bit; 5, 6, 7, or 8 data bits, depending on the data width selection; 1 PARITY bit, if parity is selected; and 1, or 2 STOP bits, depending on the STOP bit selection.

Figure 1-2 UART protocol format



Transmit/Receive for 5-bit data, parity Enable, 1 STOP bit

Transmit/Receive for 6-bit data, parity Enable, 1 STOP bit

Transmit/Receive for 7-bit data, parity Enable, 1 STOP bit

Transmit/Receive for 8-bit data, parity Enable, 1 STOP bit

1.7 Operation

1.7.1 Transmission

The UART transmitter section includes a Transmitter Buffer Register (TBR) and 16-byte FIFO. Transmitter section control is a function of the UART Line Control Register (LCR). Based on the settings chosen in LCR, The UART serializes the data and transmits the data on the UART_TXD pin.

If TX FIFO is empty and corresponding bit enable in Interrupt Enable Register (IER). An interrupt is generated. This interrupt is cleared when a character is loaded into TX FIFO.

If TX FIFO is full and corresponding bit enable in Interrupt Enable Register (IER). An interrupt is generated. This interrupt is cleared when a character sent out to UART_TXD pin and no more data loading to FIFO.

1.7.2 Reception

The UART receiver section includes a Receiver Buffer Register (RBR) and 16-byte FIFO. Receiver section control is a function of the UART line control register (LCR). Based on the settings chosen in LCR, The UART will capture the data on the UART_RXD pin.

If RX FIFO is empty and corresponding bit enable in Interrupt Enable Register (IER). An interrupt is generated. This interrupt is cleared when a character load into RX FIFO through UART_RXD pin.

If RX FIFO is full and corresponding bit enable in Interrupt Enable Register (IER). An interrupt is generated. This interrupt is cleared when data is read out by APB bus and no more data load into RX FIFO.

1.7.3 Error handling

- Access reserved address is cause HRESP response and value will be 0xFFFF_FFFF.
- Write data will be ignore when TX FIFO full.
- Read data will be undefine when RX FIFO empty.

Chapter 2 – Register Definition

2.1 Register Attribute

Table 2-1. Register attribute.

RO	Read-Only
RW	Read-Write
Rsvd	Reserved
R/W1C	Read-Write 1 to clear

2.2 Register Summary

Table 2-2. Register summary

Base offset	Name	Description
0x000	Mode Definition Register (MDR)	Sampling Mode Select
0x004	Divisor Latches Low (DLL)	Store the 8-bit low divisor for generation of the baud clock in the baud generator.
0x008	Divisor Latches High (DLH)	Store the 8-bit high divisor for generation of the baud clock in the baud generator.
0x00C	Line Control Register (LCR)	Configure UART frame, enable baud gen
0x010	Interrupt Enable Register (IER)	Enable Hardware interrupt
0x014	FIFO Status Register (FSR)	Status of RX and TX FIFO
0x018	Transmitter Buffer Register (TBR)	Transmit data to uart_txd
0x01C	Receiver Buffer Register (RBR)	Received data from uart_rxd
0x020 – 0x3FF	Reserved region	Write not affect, Read as 32’hFFFF_FFFF

2.2.1 Mode Definition Register (MDR, Offset = 0x000)

Table 2-3. Mode definition register

Bit	Name	Type	Default Value	Description
[31:1]	-	-	Rsvd	
0	OSM_SEL	RW	0	Over-sampling mode select: 1'b0 = 16× oversampling. 1'b1 = 13× oversampling.

2.2.2 Divisor Latches Low (DLL, Offset = 0x004)

Table 2-4. Divisor latches low

Bit	Name	Type	Default Value	Description
[31:8]	-	-	Rsvd	
[7:0]	DLL	RW	0	The 8 most-significant bits (LSBs) of the 16-bit divisor for generation of the baud clock in the baud rate generator

2.2.3 Divisor Latches High (DLH, Offset = 0x008)

Table 2-5. Divisor latches high

Bit	Name	Type	Default Value	Description
[31:8]	-	-	Rsvd	
[7:0]	DLH	RW	0	The 8 most-significant bits (MSBs) of the 16-bit divisor for generation of the baud clock in the baud rate generator

2.2.4 Line Control Register (LCR, Offset = 0x00C)

Table 2-6. Line control register

Bit	Name	Type	Default Value	Description
[31:6]	-	-	Rsvd	
5	BGE	RW	0	Baud generator enable 1: Enable generate. 0: Disable generate.
4	EPS	RW	0	Even parity select. Selects the parity when parity is enabled (PEN = 1) 0 = Odd parity is selected (an odd number of logic 1s is transmitted bits). 1 = Even parity is selected (an even number of logic 1s is transmitted bits).
3	PEN	RW	0	Parity enable: 0 = No PARITY bit is transmitted or checked. 1 = Parity bit is generated in transmitted data and is checked in received data between the last data word bit and the first STOP bit.
2	STB	RW	0	Number of STOP bits generated. STB specifies 1 or 2 STOP bits in each transmitted and received character. The number of STOP bits generated as below: 0 = One STOP bit is generated. 1 = Two STOP bit is generated.
[1:0]	WLS	RW	2'b11	Word length select. Number of bits in each transmitted or received serial character. 2’b00 = 5 bits 2’b01 = 6 bits 2’b10 = 7 bits 2’b11 = 8 bits

2.2.5 Interrupt Enable Register (IER, Offset = 0x010)

Table 2-7. Interrupt enable register

Bit	Name	Type	Default Value	Description
[31:5]	-	-	Rsvd	
4	en_parrity_error	RW	0	Enable paritty error Hardware interrupt
3	en_rx_fifo_empty	RW	0	Enable RX FIFO empty Hardware interrupt
2	en_rx_fifo_full	RW	0	Enable RX FIFO full Hardware interrupt
1	en_tx_fifo_empty	RW	0	Enable TX FIFO empty Hardware interrupt
0	en_tx_fifo_full	RW	0	Enable TX FIFO full Hardware interrupt

2.2.6 FIFO Status Register (FSR, Offset = 0x014)

This register is indicated status of TX FIFO and RX FIFO. Enable Hardware Interrupt to reduce polling status register by setting IER register corresponding

Table 2-8. FIFO status register

Bit	Name	Type	Default Value	Description
[31:5]	-	-	Rsvd	
4	parrity_error_status	R/W1C	0	Status of parity check for data received.
3	rx_empty_status	RO	1	Status RX FIFO empty
2	rx_full_status	RO	0	Status RX FIFO full
1	tx_empty_status	RO	1	Status TX FIFO empty
0	tx_full_status	RO	0	Status TX FIFO full

2.2.7 Transmitter Buffer Register (TBR, Offset = 0x018)

Table 2-9. Transmitter buffer register

Bit	Name	Type	Default Value	Description
[31:8]	-	-	Rsvd	
[7:0]	tx_data	WO	8’h00	Read always return 8’h00 This register can be tested in function mode, no need to register test

2.2.8 Receiver Buffer Register (RBR, Offset = 0x01C)

Table 2-10. Receiver buffer register

Bit	Name	Type	Default Value	Description
[31:8]	-	-	Rsvd	
[7:0]	rx_data	RO	8’hxx	Data is unknown in after reset This register can be tested in function mode, no need to register test

Chapter 3 – Appendixes

3.1 Basic Transmitted

Configuration basic transmit following sequence below:

1. Configure mode for UART sampling in register MDR.
2. Configure divisor value in register DLL and register DLH.
3. Enable baud generator at bit 5 (BGE) in register LCR.
4. Configure UART FRAME in register LCR.
5. Write data to buffer in register TBR.

Example for mode:

- Mode x16, baud rate 9600.
- UART FRAME: No parity, one stop bit, 8-bit data frame.
- Sequence configuration:
  - Write: address 10’h000 – data 32’h0000_0000
  - Write: address 10’h004 – data 32’h0000_008b
  - Write: address 10’h008 – data 32’h0000_0002
  - Write: address 10’h00C – data 32’h0000_0023
  - Write: address 10’h018 – data 32’h0000_0055

3.2 Basic Received

Configuration basic receive following sequence below:

1. Configure mode for UART sampling in register MDR.
2. Configure divisor value in register DLL and register DLH.
3. Enable baud generator at bit 5 (BGE) in register LCR.
4. Configure UART FRAME in register LCR.
5. Read data from buffer in register RBR after received data.

Example for mode:

- Mode x16, baud rate 9600.
- UART FRAME: No parity, one stop bit, 8-bit data frame.
- Sequence configuration:
  - Write: address 10’h000 – data 32’h0000_0000
  - Write: address 10’h004 – data 32’h0000_008b
  - Write: address 10’h008 – data 32’h0000_0002
  - Write: address 10’h00C – data 32’h0000_0023
  - Wait UART received data.
  - Read: address 10’h01C.