UART IP Core Specification

Author: Jacob Gorban
gorban@opencores.org

Rev. 0.6
August 11, 2002

This page has been intentionally left blank

Rev.	Date	Author	Description
0.1		Jacob Gorban	First Draft
0.2	27/5/01	Jacob Gorban	Added reset values and other changes.
0.3	23/6/01	Jacob Gorban	Divisor latch is 16-bit wide update
0.4	17/08/01	Jacob Gorban	Modified port names
0.5	03/12/01	Jacob Gorban	LSR bits 5,6 clear conditions fixed. In IIR, THRE was fixed. Debug registers were added to Registers. Debug interface was added to Operation. WISHBONE interface ports width modified and wb_sel_i signal is added to the list.
0.5b	10/08/02	Jacob Gorban	Added optional BAUD_O output
0.6	11/08/02	Jacob Gorban	Added parity feature implementation

Contents

1

Introduction

The UART (Universal Asynchronous Receiver/Transmitter) core provides serial communication capabilities, which allow communication with modem or other external devices, like another computer using a serial cable and RS232 protocol. This core is designed to be maximally compatible with the industry standard National Semiconductors' 16550A device.

Features:

- WISHBONE interface in 32-bit or 8-bit data bus modes (selectable)
- FIFO only operation
- Register level and functionality compatibility with NS16550A (but not 16450).
- Debug Interface in 32-bit data bus mode.

2

IO ports

2.1 WISHBONE interface signals

Port	Width	Direction	Description
CLK	1	Input	Block's clock input
WB_RST_I	1	Input	Asynchronous Reset
WB_ADDR_I	5 or 3	Input	Used for register selection
WB_SEL_I	4	Input	Select signal
WB_DAT_I	32 or 8	Input	Data input
WB_DAT_O	32 or 8	Output	Data output
WB_WE_I	1	Input	Write or read cycle selection
WB_STB_I	1	Input	Specifies transfer cycle
WB_CYC_I	1	Input	A bus cycle is in progress
WB_ACK_O	1	Output	Acknowledge of a transfer

2.2 Other internal signals

Port	Width	Direction	Description
INT_O	1	Output	Interrupt output
BAUD_O	1	Output	Optional baud rate output signal. The signal here is the 16 x actual baud rate. It is enabled if UART_HAS_BAUDRATE_OUTPUT is defined

2.3 External (off-chip) connections

Port	Width	Direction	Description
STX_PAD_O	1	Output	The serial output signal
SRX_PAD_I	1	Input	The serial input signal
RTS_PAD_O	1	Output	Request To Send
DTR_PAD_O	1	Output	Data Terminal Ready
CTS_PAD_I	1	Input	Clear To Send
DSR_PAD_I	1	Input	Data Set Ready
RI_PAD_I	1	Input	Ring Indicator
DCD_PAD_I	1	Input	Data Carrier Detect

3
Clocks

Clocks table:

Name	Source	Rates (MHz)			Description
		Max	Min	Resolution	
clk	WISHBONE bus	1258Mhz for 1200 bps	3.6864 for 115200 bps		WISHBONE clock

4
Registers

4.1 Registers list

Name	Address	Width	Access	Description
Receiver Buffer	0	8	R	Receiver FIFO output
Transmitter Holding Register (THR)	0	8	W	Transmit FIFO input
Interrupt Enable	1	8	RW	Enable/Mask interrupts generated by the UART
Interrupt Identification	2	8	R	Get interrupt information
FIFO Control	2	8	W	Control FIFO options
Line Control Register	3	8	RW	Control connection
Modem Control	4	8	W	Controls modem
Line Status	5	8	R	Status information
Modem Status	6	8	R	Modem Status

In addition, there are 2 Clock Divisor registers that together form one 16-bit.
The registers can be accessed when the 7th (DLAB) bit of the Line Control Register is set to '1'. At this time the above registers at addresses 0-1 can't be accessed.

Name	Address	Width	Access	Description
Divisor Latch Byte 1 (LSB)	0	8	RW	The LSB of the divisor latch
Divisor Latch Byte 2	1	8	RW	The MSB of the divisor latch

When using 32-bit data bus interface, additional read-only registers are available for debug purposes:

Name	Address	Width	Access	Description
Debug 1	8	32	R	First debug register
Debug 2	12	32	R	Second debug register

4.2 Interrupt Enable Register (IER)

This register allows enabling and disabling interrupt generation by the UART.

Bit #	Access	Description
0	RW	Received Data available interrupt '0' – disabled '1' – enabled
1	RW	Transmitter Holding Register empty interrupt '0' – disabled '1' – enabled
2	RW	Receiver Line Status Interrupt '0' – disabled '1' – enabled
3	RW	Modem Status Interrupt '0' – disabled '1' – enabled
7-4	RW	Reserved. Should be logic '0'.

Reset Value: 00h

4.3 Interrupt Identification Register (IIR)

The IIR enables the programmer to retrieve what is the current highest priority pending interrupt.
Bit 0 indicates that an interrupt is pending when it's logic '0'. When it's '1' – no interrupt is pending.
The following table displays the list of possible interrupts along with the bits they enable, priority, and their source and reset control.

Bit 3	Bit 2	Bit 1	Priority	Interrupt Type	Interrupt Source	Interrupt Reset Control
0	1	1	1st	Receiver Line Status	Parity, Overrun or Framing errors or Break Interrupt	Reading the Line Status Register
0	1	0	2nd	Receiver Data available	FIFO trigger level reached	FIFO drops below trigger level
1	1	0	2nd	Timeout Indication	There's at least 1 character in the FIFO but no character has been input to the FIFO or read from it for the last 4 Char times.	Reading from the FIFO (Receiver Buffer Register)
0	0	1	3rd	Transmitter Holding Register empty	Transmitter Holding Register Empty	Writing to the Transmitter Holding Register or reading IIR.
0	0	0	4th	Modem Status	CTS, DSR, RI or DCD.	Reading the Modem status register.

Bits 4 and 5: Logic '0'.
Bits 6 and 7: Logic '1' for compatibility reason.

Reset Value: C1h

4.4 FIFO Control Register (FCR)

The FCR allows selection of the FIFO trigger level (the number of bytes in FIFO required to enable the Received Data Available interrupt). In addition, the FIFOs can be cleared using this register.

Bit #	Access	Description
0	W	Ignored (Used to enable FIFOs in NS16550D). Since this UART only supports FIFO mode, this bit is ignored.
1	W	Writing a '1' to bit 1 clears the Receiver FIFO and resets its logic. But it doesn't clear the shift register, i.e. receiving of the current character continues.
2	W	Writing a '1' to bit 2 clears the Transmitter FIFO and resets its logic. The shift register is not cleared, i.e. transmitting of the current character continues.
5-3	W	Ignored
7-6	W	Define the Receiver FIFO Interrupt trigger level '00' – 1 byte '01' – 4 bytes '10' – 8 bytes '11' – 14 bytes

Reset Value : 11000000b

4.5 Line Control Register (LCR)

The line control register allows the specification of the format of the asynchronous data communication used. A bit in the register also allows access to the Divisor Latches, which define the baud rate. Reading from the register is allowed to check the current settings of the communication.

Bit #	Access	Description
1-0	RW	Select number of bits in each character '00' – 5 bits '01' – 6 bits '10' – 7 bits '11' – 8 bits
2	RW	Specify the number of generated stop bits '0' – 1 stop bit '1' – 1.5 stop bits when 5-bit character length selected and 2 bits otherwise Note that the receiver always checks the first stop bit only.
3	RW	Parity Enable '0' – No parity '1' – Parity bit is generated on each outgoing character and is checked on each incoming one.
4	RW	Even Parity select '0' – Odd number of '1' is transmitted and checked in each word (data and parity combined). In other words, if the data has an even number of '1' in it, then the parity bit is '1'. '1' – Even number of '1' is transmitted in each word.
5	RW	Stick Parity bit. '0' – Stick Parity disabled '1' - If bits 3 and 4 are logic '1', the parity bit is transmitted and checked as logic '0'. If bit 3 is '1' and bit 4 is '0' then the parity bit is transmitted and checked as '1'.
6	RW	Break Control bit '1' – the serial out is forced into logic '0' (break state). '0' – break is disabled

7	RW	Divisor Latch Access bit.
		'1' – The divisor latches can be accessed
		'0' – The normal registers are accessed

Reset Value: 00000011b

4.6 Modem Control Register (MCR)

The modem control register allows transferring control signals to a modem connected to the UART.

Bit #	Access	Description
0	W	Data Terminal Ready (DTR) signal control '0' – DTR is '1' '1' – DTR is '0'
1	W	Request To Send (RTS) signal control '0' – RTS is '1' '1' – RTS is '0'
2	W	Out1. In loopback mode, connected Ring Indicator (RI) signal input
3	W	Out2. In loopback mode, connected to Data Carrier Detect (DCD) input.
4	W	Loopback mode '0' – normal operation '1' – loopback mode. When in loopback mode, the Serial Output Signal (STX_PAD_O) is set to logic '1'. The signal of the transmitter shift register is internally connected to the input of the receiver shift register. The following connections are made: DTR ➔ DSR RTS ➔ CTS Out1 ➔ RI Out2 ➔ DCD
7-5	W	Ignored

Reset Value: 0

4.7 Line Status Register (LSR)

Bit #	Access	Description
0	R	Data Ready (DR) indicator. '0' – No characters in the FIFO '1' – At least one character has been received and is in the FIFO.
1	R	Overrun Error (OE) indicator '1' – If the FIFO is full and another character has been received in the receiver shift register. If another character is starting to arrive, it will overwrite the data in the shift register but the FIFO will remain intact. The bit is cleared upon reading from the register. Generates Receiver Line Status interrupt. '0' – No overrun state
2	R	Parity Error (PE) indicator '1' – The character that is currently at the top of the FIFO has been received with parity error. The bit is cleared upon reading from the register. Generates Receiver Line Status interrupt. '0' – No parity error in the current character
3	R	Framing Error (FE) indicator '1' – The received character at the top of the FIFO did not have a valid stop bit. Of course, generally, it might be that all the following data is corrupt. The bit is cleared upon reading from the register. Generates Receiver Line Status interrupt. '0' – No framing error in the current character
4	R	Break Interrupt (BI) indicator '1' –A break condition has been reached in the current character. The break occurs when the line is held in logic 0 for a time of one character (start bit + data + parity + stop bit). In that case, one zero character enters the FIFO and the UART waits for a valid start bit to receive next character. The bit is cleared upon reading from the register. Generates Receiver Line Status interrupt. '0' – No break condition in the current character
5	R	Transmit FIFO is empty. '1' – The transmitter FIFO is empty. Generates Transmitter Holding Register Empty interrupt. The bit is cleared when data is being been written to the transmitter FIFO. '0' – Otherwise

Bit #	Access	Description

6	R	Transmitter Empty indicator.
		    '1' – Both the transmitter FIFO and transmitter shift register are empty. The bit is cleared when data is being been written to the transmitter FIFO.
		    '0' – Otherwise

7	R	'1' – At least one parity error, framing error or break indications have been received and are inside the FIFO. The bit is cleared upon reading from the register. '0' – Otherwise.

4.8 Modem Status Register (MSR)

The register displays the current state of the modem control lines. Also, four bits also provide an indication in the state of one of the modem status lines. These bits are set to '1' when a change in corresponding line has been detected and they are reset when the register is being read.

Bit #	Access	Description
0	R	Delta Clear To Send (DCTS) indicator '1' – The CTS line has changed its state.
1	R	Delta Data Set Ready (DDSR) indicator '1' – The DSR line has changed its state.
2	R	Trailing Edge of Ring Indicator (TERI) detector. The RI line has changed its state from low to high state.
3	R	Delta Data Carrier Detect (DDCD) indicator '1' – The DCD line has changed its state.
4	R	Complement of the CTS input or equals to RTS in loopback mode.
5	R	Complement of the DSR input or equals to DTR in loopback mode.
6	R	Complement of the RI input or equals to Out1 in loopback mode.
7	R	Complement of the DCD input or equals to Out2 in loopback mode.

4.9 Divisor Latches

The divisor latches can be accessed by setting the 7th bit of LCR to '1'. You should restore this bit to '0' after setting the divisor latches in order to restore access to the other registers that occupy the same addresses. The 2 bytes form one 16-bit register, which is internally accessed as a single number. You should therefore set all 2 bytes of the register to ensure normal operation. The register is set to the default value of **0 on reset**, which disables all serial I/O operations in order to ensure explicit setup of the register in the software. The value set should be equal to (system clock speed) / (16 x desired baud rate). The internal counter starts to work when the LSB of DL is written, so when setting the divisor, write the MSB first and the LSB last.

4.10 Debug 1

This register is only available when the core has 32-bit data bus and 5-bit address bus. It is read only and is provided for debugging purposes of chip testing as it is not part of the original UART16550 device specifications. Reading from the does not influence core's bahaviour.

Bit #	Access	Description
7-0	R	Line Status Register value.
11-8	R	Interrupt Enable Register value (bits 3-0).
15-12	R	Interrupt Identifier Register value (bits 3-0).
23-16	R	Line Control Register value.
31-24	R	Modem Status Register value.

4.11 Debug 2

This register is only available when the core has 32-bit data bus and 5-bit address bus. It is read only and is provided for debugging purposes of chip testing as it is not part of the original UART16550 device specifications. Reading from the does not influence core's bahaviour.

Bit #	Access	Description
2-0	R	Transmitter FSM state
7-3	R	Number of characters in Transmitter FIFO (tf_count)
11-8	R	Receiver FSM state
16-12	R	Number of characters in Receiver FIFO (rf_count)

18-17	R	Modem Control Register value (bits 4-0)
23-19	R	FIFO Control Register value (bits 7-6)
31-24	R	Reserved. Returned value is 0.

5

Operation

This UART core is very similar in operation to the standard 16550 UART chip with the main exception being that only the FIFO mode is supported. The scratch register is removed, as it serves no purpose.

This core can operate in 8-bit data bus mode or in 32-bit bus mode, which is now the default mode.

The 32-bit mode is fully WISHBONE compatible and it uses the WISHBONE [SEL_I] signal to properly receive and return 8-bit data on 32-bit data bus. The 8-bit version might have problems in various WISHBONE implementations because a 32-bit master reading from 8-bit bus can expect data on different bytes of the 4-byte word, depending on the register address.
Also, in 32-bit data bus mode, the [ADR_I] is 5 and not 3 bits wide.

In addition, in the 32-bit data bus mode a debug interface is present in the system. This interface has 2 32-bit registers that can be read to provide non-intrusive look into the core's registers and other internal values of importance.

The selection between 32- and 8-bits data bus modes is performed by defining DATA_BUS_WIDTH_8 in uart_defines.v, uart_top.v or on the compiler/synthesizer tool command line.

5.1 Initialization

Upon reset the core performs the following tasks:

- The receiver and transmitter FIFOs are cleared.
- The receiver and transmitter shift registers are cleared
- The Divisor Latch register is set to 0.
- The Line Control Register is set to communication of 8 bits of data, no parity, 1 stop bit.
- All interrupts are disabled in the Interrupt Enable Register.

For proper operation, perform the following:

- Set the Line Control Register to the desired line control parameters. Set bit 7 to '1' to allow access to the Divisor Latches.
- Set the Divisor Latches, MSB first, LSB next.
- Set bit 7 of LCR to '0' to disable access to Divisor Latches. At this time the transmission engine starts working and data can be sent and received.
- Set the FIFO trigger level. Generally, higher trigger level values produce less interrupt to the system, so setting it to 14 bytes is recommended if the system responds fast enough.
- Enable desired interrupts by setting appropriate bits in the Interrupt Enable register.

Remember that (Input Clock Speed)/(Divisor Latch value) = 16 x the communication baud rate. Since the protocol is asynchronous and the sampling of the bits is performed in the perceived middle of the bit time, it is highly immune to small differences in the clocks of the sending and receiving sides, yet no such assumption should be made when calculating the Divisor Latch values.

6
Architecture

The core implements the WISNBONE SoC bus interface for communication with the system. It has an 8-bit data bus for compatibility reason. The core requires one interrupt. It requires 2 pads in the chip (serial in and serial out) and, optionally, another six modem control signals, which can otherwise be implemented using general purpose I/Os on the chip.
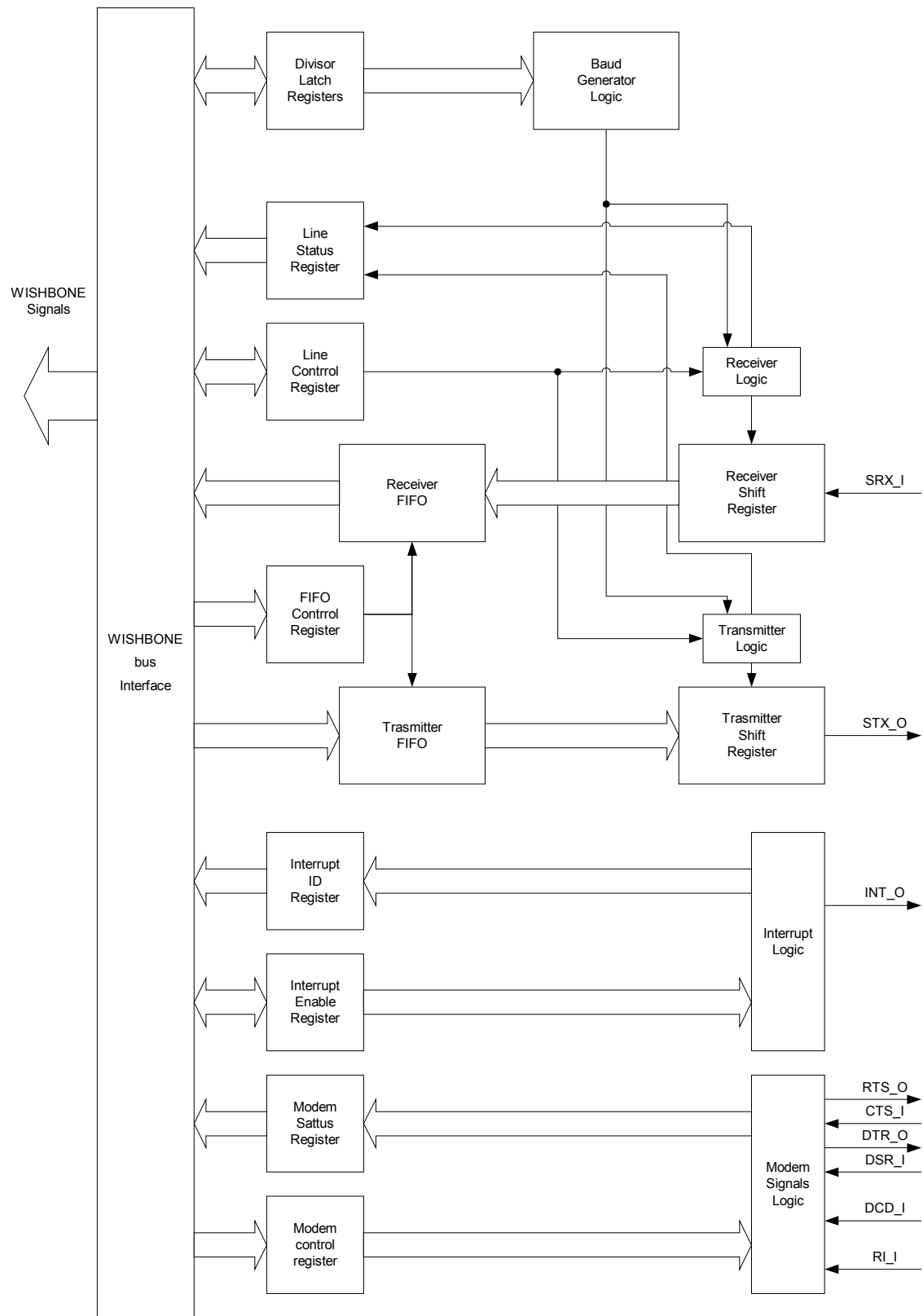
The block diagram of the core is on the following page.

Figure 1 - Block Diagram of the UART core