International Journal of VLSI System Design and Communication Systems

ISSN 2322-0929
Vol.03, Issue.03,
June-2015,
Pages:0262-0264

www.ijvdcs.org

Design and Development of Verification Environment to Verify UART Protocol using UVM

NIRANJAN B S[1], SUBODH KUMAR PANDA[2], SHANTHI V A[3]

[1]PG Scholar, Dept of ECE, B.N.M Institute of Technology, Bengaluru, India, E-mail: niranjanbs.das@gmail.com.
[2]Assoc Prof, Dept of ECE, B.N.M Institute of Technology, Bengaluru, India, E-mail: subodhpanda2013@gmail.com.
[3]Senior Technical Staff, Maven Silicon Softech Pvt Ltd, Bengaluru, India, E-mail: shanthi@maven-silicon.com.

Abstract: The main objective of the work is to design an UART Protocol using Verilog HDL and Verify the designed UART Protocol using Universal Verification Methodology In this era of electronics the UART (Universal Asynchronous Receiver/Transmitter) playing an important role in data transmission. This UART protocol provides serial communication capabilities, which allows communication with modems or other external devices. This protocol is designed to be maximally compatible with industry standard designs. The key features of this design are Wishbone Interface with 8-bit or 32-bit selectable data bus modes. Debug interface in 32-bit data bus mode. Register level and functional compatibility. FIFO operation. UART protocol can be designed using Verilog HDL and Synthesized using Xilinx 13.2, and then can be simulated using Questasim 10.0b. Test bench is written with regression test cases in order to acquire maximum functional coverage.

Keywords: UART, Wishbone, Questasim, Xilinx ISE, Verilog, UVM, Coverage.

I. INTRODUCTION

The data transmission takes place in-between the systems, in between the chips and inside the chips also. As it is asynchronous clock there will be no method to set up the clock distribution techniques. There are three of modes in UART. They are

Half Duplex Mode: In the half duplex mode either reception or transmission takes place at a time. 2) FULL DUPLEX MODE: In the full duplex mode both reception and transmission takes place at a time. 3) LOOP BACK MODE: This mode is used for testing purpose. We will be connecting the receiver and transmitter of same UART this helps to check the accuracy of it.

II. UART DESIGN PRINCIPLES

Verilog HDL is one of the hardware description languages. Verilog can be used for different levels of logic design, it can be used for digital system logic simulation, timing analysis and logic synthesis. In this work, a UART Protocol Module is designed using Verilog HDL.

A. Architecture of UART

The architecture of UART mainly consists four building blocks, they are

- Wishbone Interface
- Interrupt Registers
- Baud Generator
- Divisor Latches

The architecture of UART IP Core is as shown in fig.1.

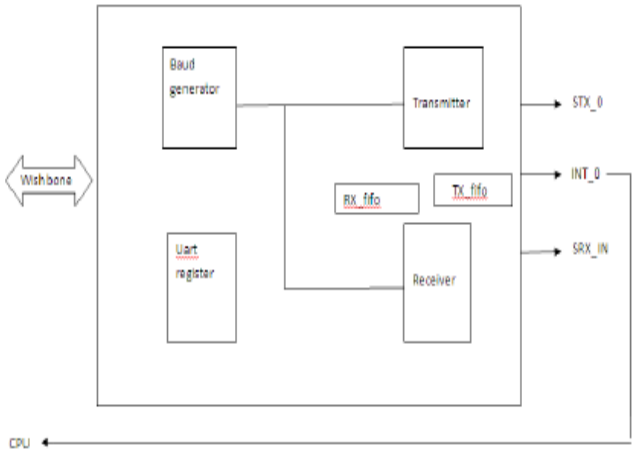

Fig.1. Architecture of UART.

Wishbone Interface: This is the standard computer bus interface which allows communicating between the integrated circuits. The wishbone bus permits both 8-bit and 32-bit data transfer. Wishbone Interface signals are shown in fig.2.

Interrupt Registers: Interrupt Register is used to enable and identify interrupts. There are 2 types of interrupt registers. They are Interrupt Identification Register and Interrupt Enable Register. The interrupt enable register enables and also disables with interrupt generation by the UART IP Core. It's of 8-bit width. The interrupt identification register enables the programmer to find the current highest priority

pending interrupt. BIT-0 shows that an interrupt is pending when it is logic"0". When it is"1" it shows that the interrupt is not in pending. It's of 8-bit width.

PORT	WIDTH	DIRECTION
CLK	1	INPUT
WB_RST_I	1	INPUT
WB_ADDR_I	5or3	INPUT
WB_SEL_I	4	INPUT
WB_DATA_I	32or8	INPUT
WB_WE_I	1	INPUT
WB_STB_I	1	INPUT
WB_CYC_I	1	INPUT
WB_DATA_O	32 or 8	OUTPUT
WB_ACK_O	1	OUTPUT

Fig.2. Wishbone Interface signals.

Control Registers: There are two Control registers. They are FIFO control register and LINE control register. The FIFO control register permits selection of the FIFO trigger level. It is of 8-bit data width. The LSB bit should be always "0". The line control register permits the specification of the format of the asynchronous data communication used. A bit in the register also permit access to the divisor latches, which defines the baud rate. Reading from the register is permitted to check the current setting of the data communication.

Baud Generator: It is responsible for a periodic baud pulse generation based on the divisor latch value which determines the baud rate for the serial transmission. This periodic baud pulse is used by receiver and transmitter to generate sampling pulses for sampling both data to be transmitted and received data. One baud out occurs for sixteen clock cycles. One bit data will be sent for sixteen clock cycles. There are two debug registers and they work in 32-bit data bus mode. It has 5-bit address mode. It is read only and it is provided for debugging purpose for chip testing. Each has a 256-byte FIFO to buffer data flow. The FIFO buffers increases the overall transmission rate by permitting slower processors to respond, and reducing the amount of time wasted context switching. Besides data transfer, they also facilitate start/stop framing bits, check various parity options, and also detects transmission errors.

Divisor Latches: The divisor latches can be accessed by setting the 7th bit of LCR. Restore the LCR 7th bit to zero after setting the divisor latches in order to restore access to the other registers that occupy the same address. The two bytes form one sixteen bit register, which is internally accessed as a single number. In order to have normal operation two bytes are driven to zero on reset. The reset 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*desired baud rate. The internal counter starts to work when the LSB of DL is return, so when setting the divisor, write the MSB first then LSB last.

B. Top Module of UART

We get the basic structure UART Protocol by including the Wishbone Interface module and UART registers module. An important aspect of the top-level module is to ensure that the sub-modules work smoothly. The UART top module needs the control word, the efficient operation of sub-modules: Wishbone Interface module and UART registers module. UART top module is synthesized by ISE. Synthesized module is shown in Fig.3.

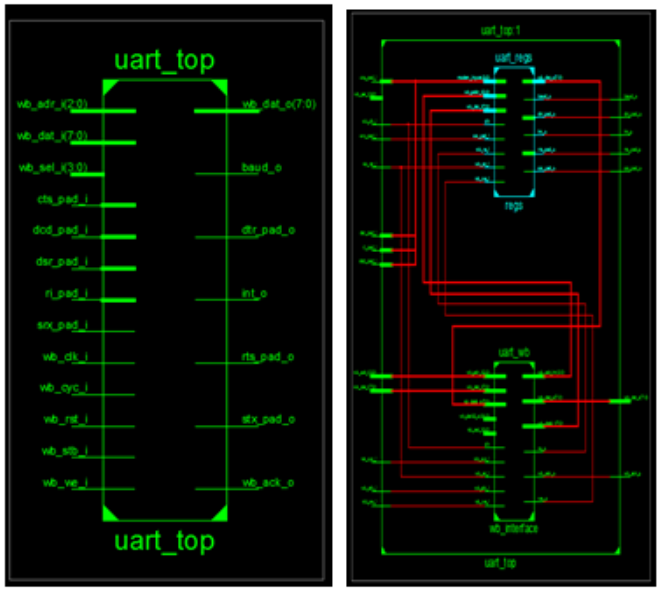

Fig.3. UART Top module.

III. SIMULATION AND VERIFICATION

Verilog HDL is one of the hardware description languages. Verilog can be used for different levels of logic design, it can be used for digital system logic simulation, timing analysis and logic synthesis. In this work, a UART Protocol Module is designed using Verilog HDL. Using Verilog HDL we had designed the UART Protocol circuit, synthesised with ISE, and then used Questasim to simulate. In order to establish the test platform, first setup the master module to simulate the Wishbone protocol, and then setup the slave module to simulate UART protocol at the same time. Then to compare and check the receive/send data and addresses. The test bench is designed using universal verification methodology. Verification is also the major part of this work as many modules are designed and having many inputs universal verification methodology helps to makes easy to stimulate according to our wish, constraint the unwanted stimulus. Here the UART Protocol Design is considered as design under test (DUT). Components of

Verification Environment are as follows : RTL module, Master and Slave agent tops which consists of master and slave agents which contains sequencer, driver and monitor, virtual sequencer, Scoreboard, TOP level module which encapsulates the above all with RTL and Test Cases. UVM test bench architecture is as shown in Fig.4.

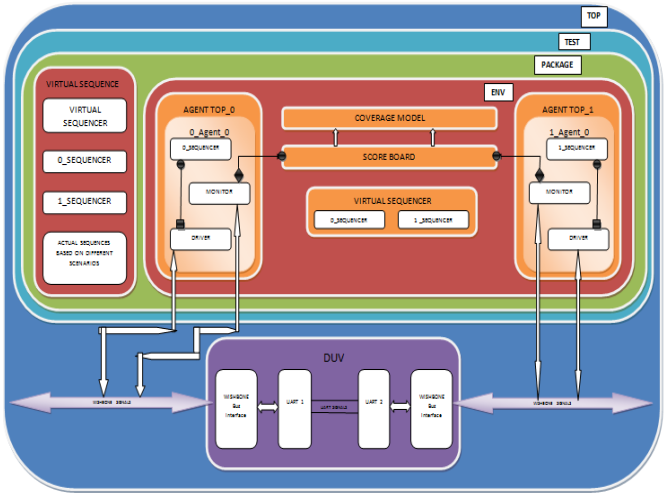

Fig.4. UVM testbench Architecture.

The simulation results are as shown below in Fig.5.

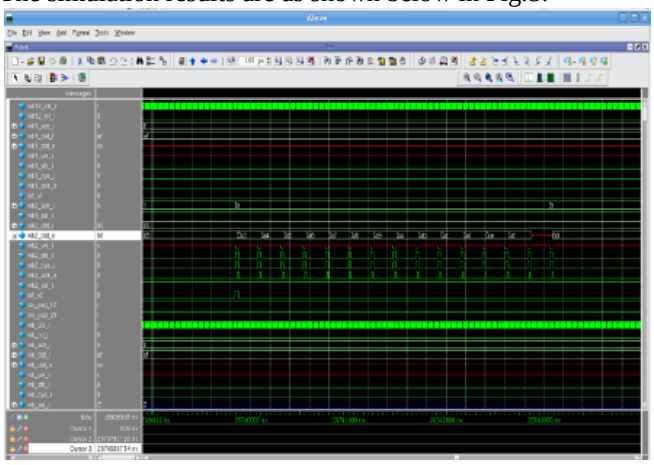

Fig.5. Simulation results.

Coverage Report is shown below in Fig. 6

Number of tests run:	7
Passed:	7
Warning:	0
Error:	0
Fatal:	0

List of tests included in report...

Coverage Summary by Structure:

Design Scope	Coverage (%)
top	100.00%
agt0	100.00%
agt1	100.00%
uvm_pkg	100.00%
uvm_callbacks	100.00%
uvm_phase	100.00%
uvm_component	100.00%
uart_test_pkg	100.00%
uart_1xtns_agt0	100.00%
uart_2xtns_agt0	100.00%
uart_3xtns_agt0	100.00%
uart_4xtns_agt0	100.00%
uart_5xtns_agt0	100.00%

Coverage Summary by Type:

Weighted Average:				100.00%
Coverage Type	**Bins**	**Hits**	**Misses**	**Coverage (%)**
Covergroup	96	96	0	100.00%
Assertion Attempted	128	128	0	100.00%
Assertion Failures	128	0	-	0.00%
Assertion Successes	128	128	0	100.00%

Fig.6. Coverage report.

IV. CONCLUSION

In this work, we have designed the UART Protocol Using Verilog HDL language based upon design-reuse methodology. The UART IP core working has been verified for different modes of operations i.e. HALF DUPLEX MODE , FULL DUPLEX MODE and LOOPBACK MODE.. Further, we have also done functional verification. The code coverage is obtained for the RTL design and 100% code coverage and functional coverage is extracted. The complete function of the registers is verified in this paper. The innovation of this article: Universal verification methodology is used in order to cover all the functions of the code and to reuse the class instead of writing the code again. The advantage of using universal methodology is making the things easy by calling the inbuilt classes.

V. REFERENCES

[1] I.E.Sutherl and 'Micropiplines' Communication ACM, June 1989, Vol. 32(6), pp. 720 -738.
[2] V.N. Yarmolik, Fault Diugnosis of Digital Circuits, JohnWiley & Sons, 1990.
[3]"PCI6550D Universal Asynchronous Receiver/ Transmitter with FIFOs", National Semiconductor Application Note, June 1995.
[4] M.S.Harvey, "Generic UART Manual" Silicon Valley. December 1999.
[5] Martin S. Michael, "A Comparison of the INS8250,NS16450 and NS16550AF Series of UARTs" National Seiniconductor Application Note 493, April1989.
[6]W.Elmenreicb,M.Delvai, Time Triggered Communication with UARTs. In Proceedings of the 4'h IEEE International Workshop on Factory Communication Systems, Aug. 2002.
[7] UVM from Cookbook 2012.
[8] A verilog HDL primer, Star galaxy Press, 1997.
[9] A Verilog Synthesis Primer, Star galaxy press, 1997.
[10] UART IP Core Specification, Jacob Gorban.