International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 2, February - 2014

Architectural Low Power Implementation of UART using Verilog

Mohammed Azeemuddin, Prof. S.K Ahmeduddin Zakir
Shadan College of Engineering and Technology Hyderabad, India

Abstract - UART is Universal Asynchronous Receiver Transmitter. It is mostly used for short-distance, low speed, low-cost data exchange between peripherals. The UART allows the devices to communicate without need to be synchronized. In this project we try to minimize the power consumption by using low power techniques. With the proliferation of portable electronic devices, power efficient data transmission has become increasingly important. Components such as laptop, keyboards, palmtops and modems are few examples of devices that employ UART circuits. In this project we are enhancing IEEE paper in which a normal UART is synthesized and results are simulated using Xilinx then these files are run under Power compiler tool to find power consumption by normal UART and then by using two techniques of low power we reduce power consumption which is shown in the power reports. The UART is implemented using Verilog which is having more advantages as compared to VHDL.

1. INTRODUCTION

An UART (universal asynchronous receiver / transmitter) is responsible for performing the main task in serial communications with computers. The device changes incoming parallel information to serial data which can be sent on a communication line. A second UART can be used to receive the information. The UART performs all the tasks, timing, parity checking, etc. needed for the communication. The only extra devices attached are line driver chips capable of transforming the TTL level signals to line voltages and vice versa.

To use the device in different environments, registers are accessible to set or review the communication parameters. Setable parameters are for example the communication speed, the type of parity check, and the way incoming information is signaled to the running software.

The serial port on your PC is a full-duplex device meaning that it can send and receive data at the same time. In order to be able to do this, it uses separate lines for transmitting and receiving data. Some types of serial devices support only one-way communications and therefore use only two wires in the cable - the transmit line and the signal ground.

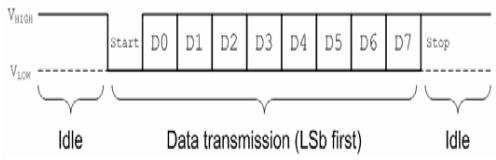

Figure 1 Format of data transfer in UART

Once the start bit has been sent, the transmitter sends the actual data bits. There may either be 5, 6, 7, or 8 data bits, depending on the number you have selected. Both receiver and the transmitter must agree on the number of data bits, as well as the baud rate. Almost all devices transmit data using either 7 or 8 databits. As shown in figure start bit is low and stop bit is high, when data bits plus start bit is transferred counter identifies number of data bits then checks for stop bit and when encountered it will stop transmission.

Notice that when only 7 data bits are employed, you cannot send ASCII values greater than 127. Likewise, using 5 bits limits the highest possible value to 31. After the data has been transmitted, a stop bit is sent. A stop bit has a value of 1 - or a mark state - and it can be detected correctly even if the previous data bit also had a value of 1. This is accomplished by the stop bit's duration. Stop bits can be 1, 1.5, or 2 bit periods in length.

2. IMPLEMENTATION

In this paper, the top to bottom design is used. The UART serial communication is divided into three sub-modules: the baud generator, receiver module and transmitter module. Therefore, the implementation of the UART communication module is actually the realization of the three sub-modules. The baud generator is used to produce a local clock signal which is much higher than the baud rate to control the UART receive and transmit module. The receiver module is used to receive the serial signals and convert them into parallel data. The UART transmit module converts the bytes into serial bits according to the basic frame format and transmit those bits.

Baud Generator

Here we want to use the serial link at maximum speed, i.e. 115200 bauds. Other slower speeds would also be easy to generate. It is responsible to generate bauds for the transmitter and receiver modules one is 16 times other is 8 times.

FPGAs usually run at speed well above 115200Hz (RS-232 is pretty slow by today's standards). That means we use a high-speed clock and divide it down to generate a "tick" as close as possible to 115200 times a second.

Transmitter Module

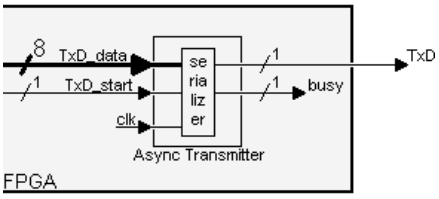

Figure 2 Transmitter module

We are building an "async transmitter" and it works like that

- The transmitter takes 8-bits data, and serializes it (starting when the "TxD_start" signal is asserted).
- The "busy" signal is asserted while a transmission occurs. The "TxD_start" signal is ignored during that time.

The RS-232 parameters used are fixed: 8-bits data, 2 stop bits, no-parity.

Receiver Module

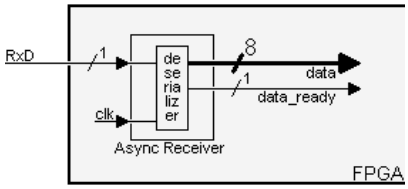

Figure 3 Receiver Module

We are building an "async receiver" and our implementation works like that:

- The module assembles data from the RxD line as it comes.
- As a byte is being received, it appears on the "data" bus. Once a complete byte has been received, "data_ready" is asserted for one clock.

An asynchronous receiver has to somehow get in-sync with the incoming signal (it doesn't have access to the clock used during transmission).

- To determine when a new data is coming ("start" bit), we oversample the signal at a multiple of the baud rate frequency.

Once the "start" bit is detected, we sample the line at the known baud rate to acquire the data bits.

3. LOW POWER TECHNIQUES USED

In this paper we have used two techniques that is voltage scaling and Module Powerdown, apart from these two we have many techniques like Clock gating, MT-CMOS, LT-ASCII and Data stream dependent shutdown.

Voltage Scaling

In this technique we are going to reduce the supply voltage itself which will reduces the power consumption as power is related to supply voltage as shown in below equation.

$$P \approx \alpha\, f_{clk}\, C_L\, V_{dd}^2$$

Where α is the activity factor, f_{clk} is the operating frequency, C_L is the load capacitance and V_{dd} is the supply voltage.

Module Powerdown

In this technique we use to turn off the module like receiver when baud generator and transmitter are working and viceversa. As shown in above equation again power is directly proportional to square of voltage and this voltage is equal to sum of the all three module voltages.

4. RESULTS AND DISCUSSIONS

After synthesizing the top module of UART in Xilinx software we get the schematic of the RTL as shown in below diagram.

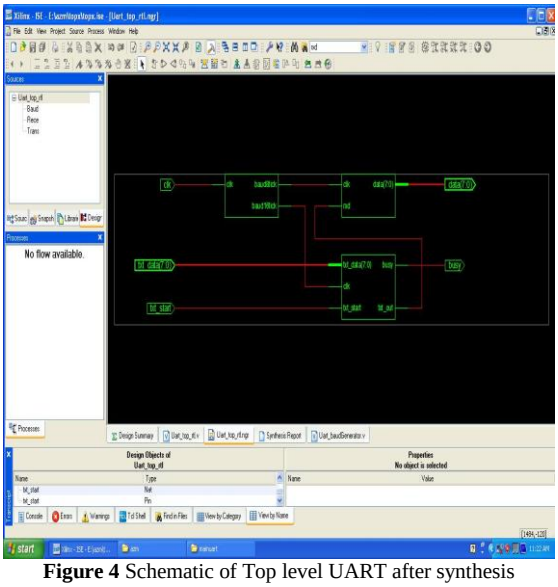

Figure 4 Schematic of Top level UART after synthesis

5. SIMULATION

After synthesizing the code let us now simulate the baud generator in a Xilinx Simulator, the waveform is as shown below which produces two bauds one for transmitter module and another for receiver module which is oversampled.

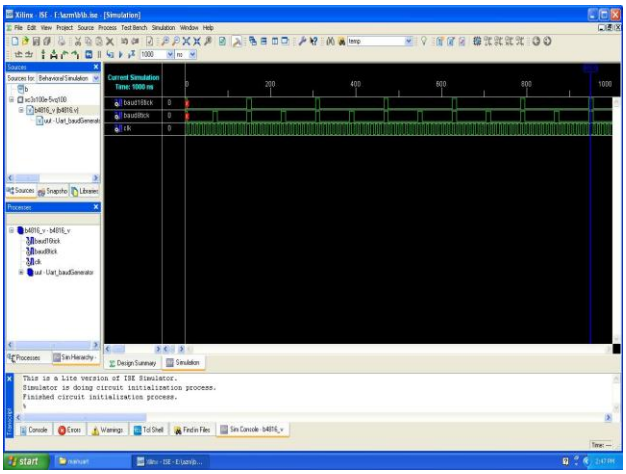

Figure 5 waveform of Baud Generator

Next is transmitter module to which parallel input is applies and it produces serial output which can be seen in the below simulation result.

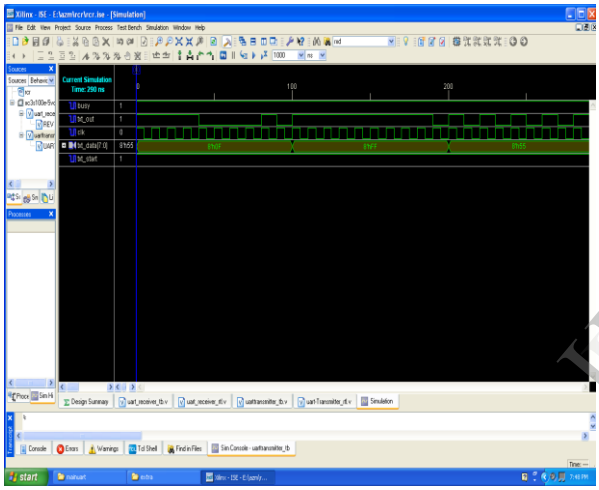

Figure 6 waveform of Transmitter Module

At the Receiver side the serial data is acting as an input and the output of this will be parallel data as shown in below simulation waveform.

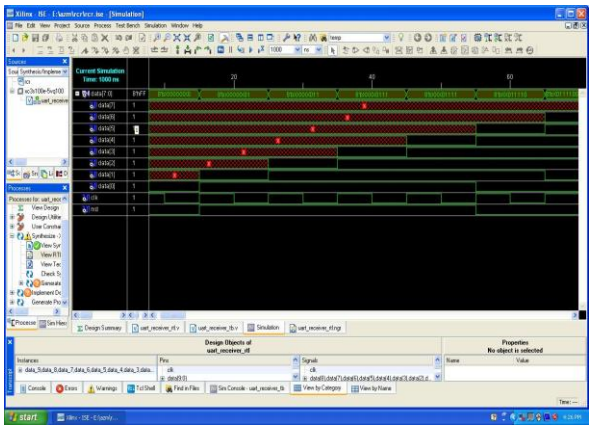

Figure 7 Waveform of Receiver Module

Table 1 Comparison of normal UART with Low power UART

MODULES USED	POWER CONSUMPTION IN mW	%OF POWER SAVED
NORMAL UART TOP	13.124	--
LOW POWER UART TOP	2.028	84.54
BAUDGEN & TRANSMITTER(LOW)	1.42	89.18
BAUDGEN & RECEIVER(LOW)	1.38	89.48

Power Report of Top UART (low)

We run the top module Verilog files in the power compiler tool by reducing power supply voltage and it generates power reports.

```
Design                  Wire Load Model          Library
------------------------------------------------------------------------------

uart_top_low            suggested_10K    umcl18g212t3_tc_120V_25C
Global Operating Voltage = 1.2 V
Power-specific unit information:
  Voltage Units = 1V
  Capacitance Units = 1.000000pf
  Time Units = 1ns
  Dynamic Power Units = 1mW    (derived from V,C,T units)
  DynamicPower =  1.876 mW   (92.51%)
  Static Power =  0.152mW   (7.49%)
                  ----------
  Total Power   = 2.028  mW  (100%)
```

Figure 8 Power report of Top UART (low)

5. CONCLUSION

- Several low power configurations were examined to minimize the UART power consumption. But for our project we used only two techniques that is voltage scaling and module powerdown because power analyzation is quite easy in this case.
- No doubt other techniques are also very useful but we are restricted to academically possible techniques.
- With voltage scaling practically we saved approximately 84% of power and in case of module power down it is approximately 89% of power consumption is reduced.
- Future of this UART also depends on the advancement in low power design which may reduce much more power consumption.

6. REFERENCES

1. IEEE paper: Design & simulation of UART serial communication module based on VHDL by Fang yi yaun and Chen Xue-jun, Shanghai universityof science China, ISA 28-2 May 2011.
2. CMOS VLSI DESIGN- A Circuits and Systems Perspective by Neil H.E. Weste, David Harris, Ayan Banerjee. 3rd Edition.
3. Verilog HDL –A Guide to Digital Design and Synthesis by Samir Palnitkar, Sunsoft Press 1996.
4. A.P. Chandrakasan, S. Sheng and R.W. Brodersen, "Low Power CMOS Digital Design". IEEE J. Solid-State circuits, Vol.27, no.4, Apr.1992,pp. 473-484.
5. S. Mutoh, T.Douseki, Y.Matsuya, T. Aoki, S.Shigematsu and J.Yamada, "1-V power supply High Speed Digital Circuit Technology with Multi Threshold – Voltage CMOS" IEEE J. Solid-State circuits, August1995,pp.847-854.
6. D.Liu, and C.Svensson, "Trading Speed for Low Power by Choice of Supply and Threshold Voltages". IEEE J. Solid-State circuits, Vol.28, no.1, Jan.1993, pp.10-17.
7. P. Horowitz, W. Hill, "The Art of Microelectronics", Second edition, pp. 711-812.
8. www.fpga4fun.com
9. A. P. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," Proceedings of the IEEE, vol. 83, no. 4, April 1995, pp. 498-523.
10. Q. Wu, M. Pedram, and X. Wu, "Clock Gating and its Applications to Low Power Design of Sequential Circuits," IEEE Transactions on Circuits and Systems, vol. 47, no. 103, March 2000, pp. 415-420.
11. R. Gonzalez, B. Gordon, and M. A. Horowitz, "Supply and Threshold Voltage Scaling for Low Power CMOS," IEEE Journal of Solid-State Circuits, vol. 32, no. 8, August 1997, pp. 1210-1216.
12. Jan M Rabey and Massoud Pedram (Edited)," Low Power Design Methodologies", Kluwer Academic Publishers, 1996.
13. Dughals A Pucknell and Kamran Eshraghain,"Basic VLSI Design,", Third edition, Prentice-Hall of India Pvt Ltd,2001.
14. Neil E Weste and David Harris,"CMOS VLSI Design A Circuits and Systems Perspective", Third Edition, Pearson Education, 2005.
15. Wayne Wolf, "Modern VLSI Design "Pearson Education, 3rd edition 2005.