Global Journal of Researches in Engineering
Electrical and Electronics Engineering
Volume 12 Issue 7 Version 1.0 July 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

A Qualitative Approach to Design Multi Channel UART Using FPGA and FIFO Technologies

By C.K.Hemantha Lakshmi, C.K.Hemantha Rama & C.K.Mahesh Babu

Dept. of ECE, MITS, Madanapalle, Andhra Pradesh

Abstract - To meet modern complex control systems communication demands, my paper presents a multi channel UART controller based on FIFO (first in first out) and FPGA (field programmable gate array).The paper presents design method of asynchronous FIFO and structure of controller.This controller is designed with FIFO circuit block and UART (universal asynchronous teceiver transmitter) circuit block with in FPGA to implement communication in modern complex control systems quickly and effectively. This controller can be used to implement communication when master equipment and slaver equipment are set at different baud rate. It can also be used to reduce synchronization error between sub systems in asystem with several sub systems. The controller is reconfigurable and scalable. The whole scenario is simulated using VHDL coding.

GJRE-F Classification : FOR Code: 090601

Strictly as per the compliance and regulations of:

A Qualitative Approach to Design Multi Channel UART Using FPGA and FIFO Technologies

C.K.Hemantha Lakshmi [α], C.K.Hemantha Rama [σ] & C.K.Mahesh Babu [ρ]

Abstract - To meet modern complex control systems communication demands, my paper presents a multi channel UART controller based on FIFO (first in first out) and FPGA (field programmable gate array).The paper presents design method of asynchronous FIFO and structure of controller.This controller is designed with FIFO circuit block and UART (universal asynchronous teceiver transmitter) circuit block with in FPGA to implement communication in modern complex control systems quickly and effectively. This controller can be used to implement communication when master equipment and slaver equipment are set at different baud rate. It can also be used to reduce synchronization error between sub systems in asystem with several sub systems. The controller is reconfigurable and scalable. The whole scenario is simulated using VHDL coding.

I. Introduction

Today, owing to availability of state-of–the–art microcontrollers and digital signal processors(DSPs), complex control algorithms can be easily implemented to attain the desired system performance [3].But in actual control systems it is difficult to attain the expected result for various factors affect the control systems such as control algorithms itself, capability of implement equipment and states of control circumstance. Except those factors, communication parameters of control systems including baud rate, BER (bit error rate) and synchronization between sub systems also engender great effect .In order to improve precision of control system and make good use of modern control algorithms ,we should pay much more attention on communication in control systems[1].

In several control systems, UART a kind of serial communication circuit used widely. A universal asynchronous receive transmit is an integrated circuit which plays most important role in serial communication. It handles the conversion between serial and parallel data. Serial communication reduces the distortion of a signal therefore makes data transfer between two systems separated in great distance possible[2].

In some complex systems, communications between the master controller and slaver controllers are implemented by serial or parallel port. Parallel communication needs a lot of multi-bit address bus and data bus and it is only convenient for short distance transmission. Serial communication is another way of communication used extensively because of its simple structure and long transmission distance .But sometimes a common serial port could not meet requirements of complex systems with different baud rates equipments even some special baud rate equipments. It is impossible to implement this multi baud rate communication system without a special baud rate converter[3].

a) Proposed system is

- RTL design and verification of RS 232.
- Interfacing digital clock manager.

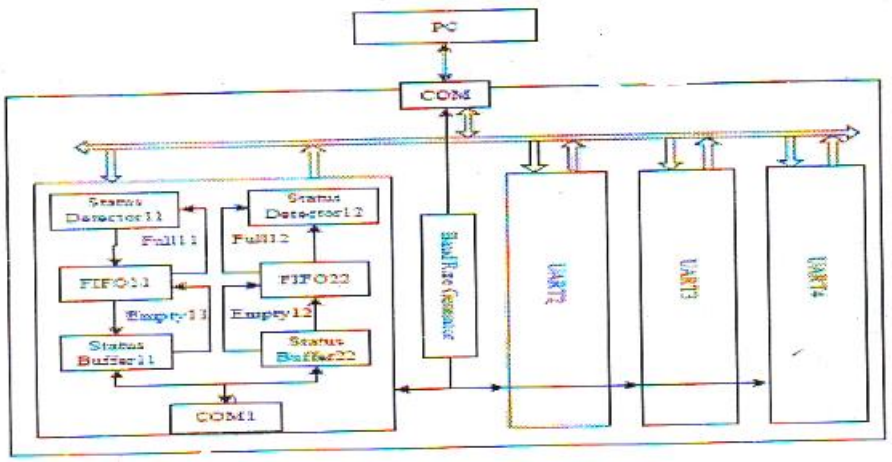

Figure 1 : Multi Channel UART

Author α : Assistant Professor, Dept., of ECE, MITS, Madanapalle, Andhra Pradesh. E-mail : hemanthaster@gmail.com
Author σ : Assistant Professor, Dept., of CSE, KEC, Kuppam, Andhra Pradesh. Email : ckhemantha@gmail.com
Author ρ : M.Tech, Dept of EEE., NIT Nagpur, Email : ckmahesh111@gmail.com

II. Design Implementation

a) Designing of Asynchronous FIFOs

An asynchronous FIFO refers to a FIFO design where data values are written to a FIFO buffer from one clock domain and data the data value are read from the same FIFO buffer from another clock domain, where the two clock domains are asynchronous to each other. FIFOs are always used for data cache, storing differences of frequency or phase of asynchronous signals. Asynchronous FIFOs are often used to quickly and safely pass data from one clock domain to another asynchronous clock domain. In asynchronous clock circuit, periods and phases of each clock domain are completely independent so the probability of data loss is always not zero.

b) Structure of Asynchronous FIFO

A FIFO consists of a RAM array block status block writer pointer (WR_ptr) and a read pointer (RD_ptr) and its structure is shown below figure

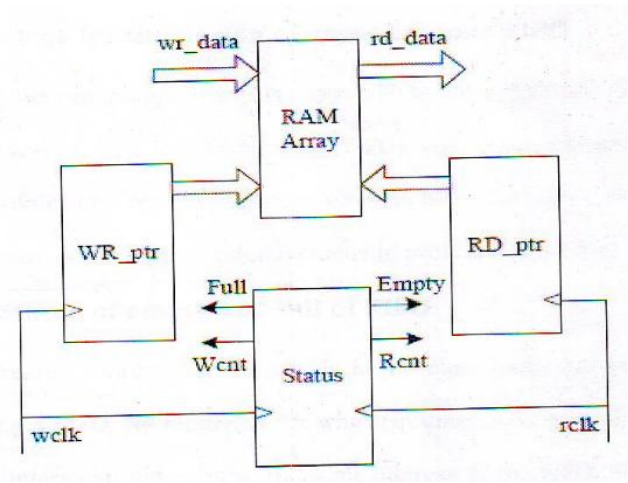

Figure 2 : Asynchronous FIFO

A RAM array with separate read and write ports is used to store data .The writer pointer points to the location that will be written next, and read pointer points to the location that will be read currently. A write operation increments the writer pointer and read operation increments the read pointer. On reset, both pointers are rest to zero, the FIFO is empty. The write pointer happens to be the next FIFO location to be written and read pointer is pointing to invalid data. The responsibility of status block is to generate the EMPTY and FULL signals to the FIFO [4]. If the FULL signal is active then the FIFO cannot accommodate more data and if the EMPTY is active then the FIFO cannot provide more data to read out. When writing data into the FIFO „wclk"will be used as the clock domain and when reading data out of the FIFO „rclk" will be used as the clock domain. These both clock domains are asynchronous.

c) Factors for the design of Asynchronous FIFO

In the designing of asynchronous FIFOs, two important factors are considered. One is how to judge FIFOs status according to the writer pointer and read pointer. The other is how to design circuit to synchronize asynchronous clock domains to avoid metastability.

d) Status of empty and full of FIFO

Creating empty and full signals is the most important part of designing a FIFO. No matter under what circumstance, the read and write pointers cannot point to the same address of the FIFO. So the empty and full signals play very important roles within FIFO that they block access to further read or write respectively. Generally, in an ordinary FIFO, when the read pointer equals the write pointer the FIFO is empty. But in a circular FIFO it is either empty or fll when both of the pointers are equal. Because the FULL and EMPTY signals can not only be decided by the pointers value but also influenced by the operation that caused the pointers to become equal. If a reset or read makes the pointers equal to each other, the FIFO is really empty. If a write makes the pointers equal, the FIFO is full.

One design technique used to distinguish between FULL and EMPTY is to add an extra bit to each pointer. When the write pointer increments past the final FIFO address, the write pointer will increment the unused MSB while setting the rest of the bits back to zero as shown In below figure(the FIFO has wrapped and toggled the pointer MSB).The same is done with the read pointer. If the MSBs of the two pointers are different, it means that the write pointer has wrapped one more time that the read pointer. If the MSBs of the two pointers are the same, it means that both pointers have wrapped the same number of times.

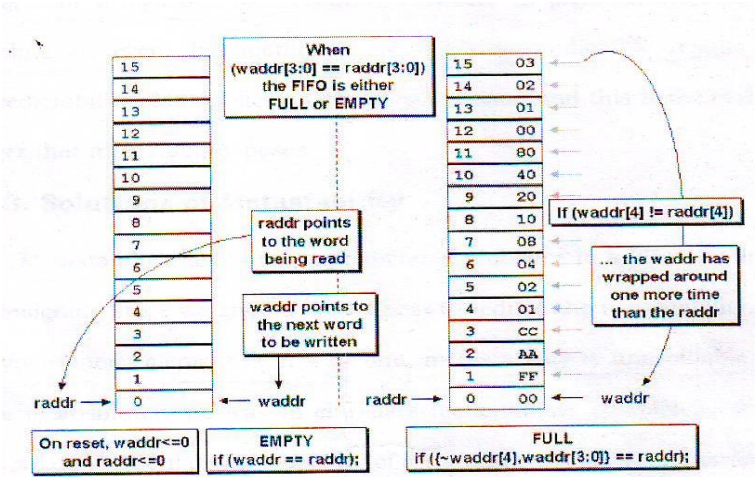

Figure 3 : FIFO organization

Using n-bit pointers where (n-1) is the number of address bits required to access the entire FIFO memory buffer. As shown in above figure the FIFO is empty when both pointers, including the MSBs are equal. And the FIFO is full when both pointers. Except the MSBs are equal. The FIFO design described here uses n-bit pointers for a FIFO with 2(n-1) write-able locations to hep handle FULL and EMPTY conditions.

e) UART circuit block and structure

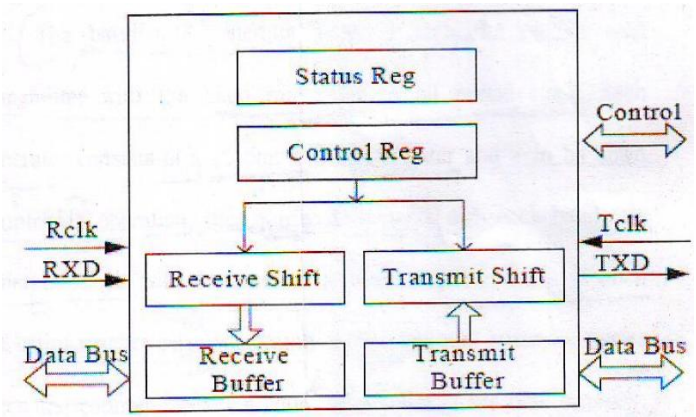

Figure 4 : UART Block diagram

f) Software structure

We can use software codes in VHDL to design FPGAs hardware structure, it is easy to create and adjust to satisfy requirements of applications. There are one UART used to communicate with PC or other main MCU and there are also four other UARTs used to communication with sub MCUs [5]. Each channel has two FIFOs one for receiving data and the other for transmitting data. Each FIFO"s depth is 64 bytes. The software flow chart is shown in figure 6.

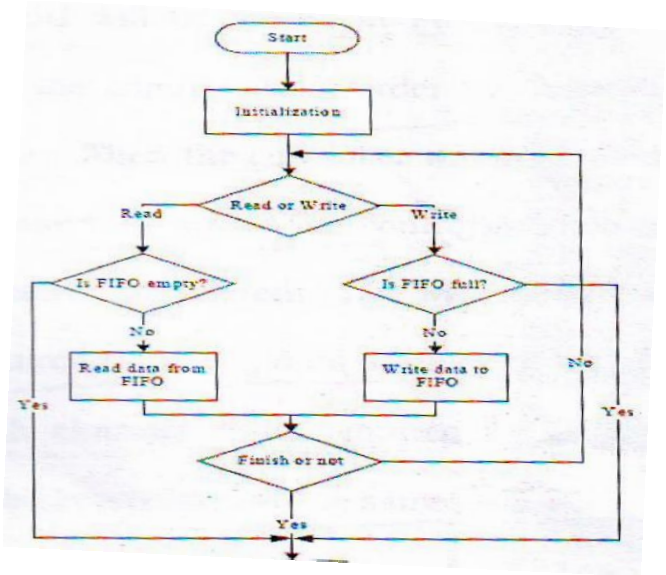

Figure 5 : Proposed method Flow Chart

As shown in figure ,when FIFO is full we cannot write any more byte into the FIFO.At this time, the status detector will set CS high to indicate that the FIFO is full and when FIFO is full you cannot write any more byte into the FIFO. At this time, the status detector will set CS high to indicate that the FIFO is full and stop writing to the FIFO. When FIFO is empty we cannot read from it any more. Then the status detector will set empty high to indicate the status of FIFO and stop reading from it [6][7].

When FIFO is not full or empty it will be written or read data according to the control order. After finishing all write or read operation it will stop until next access is coming.

III. Simulation Results

a) Simulation And Verification

To verify design of the controller a test bench is written to make verification in modelsim .Data received from the PC or other main MCU will be stored in FIFO"s within FPGA till the controller will set a kind of baud rate according to commands desired. The controller is receiving data and store the data received to different FIFO waiting for read. When sub-controllers are required to receive data at different baud rates, the controller can set each channel at its required baud rate to transmit data. The controller sends data at the same time but at different baud rate.

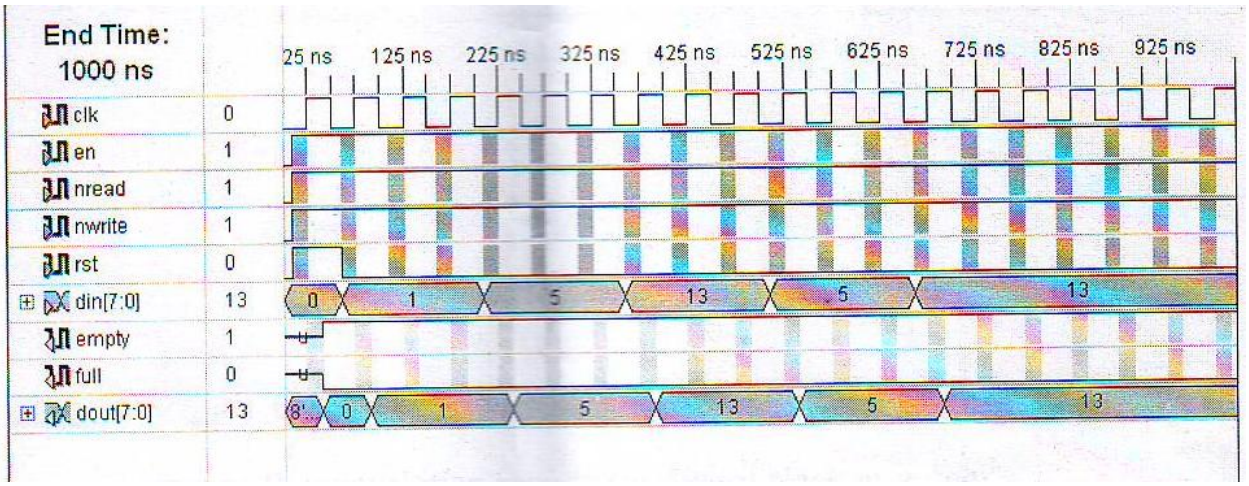

Figure 6 : Simulation results for single FIFO

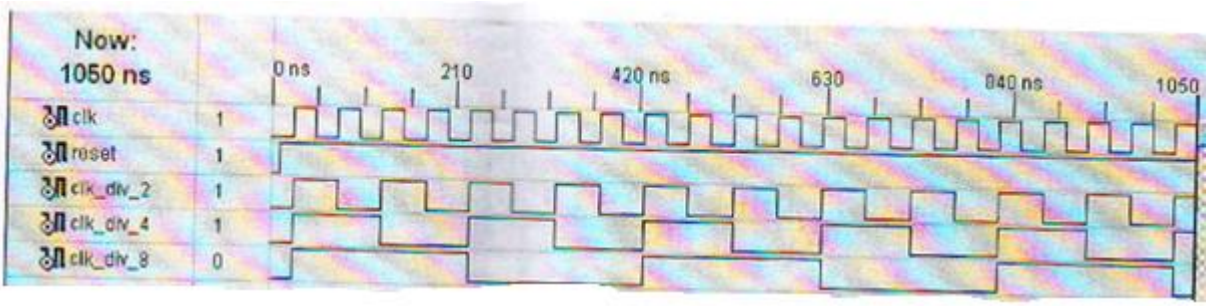

Figure 7 : Simulation results for digital clock manager

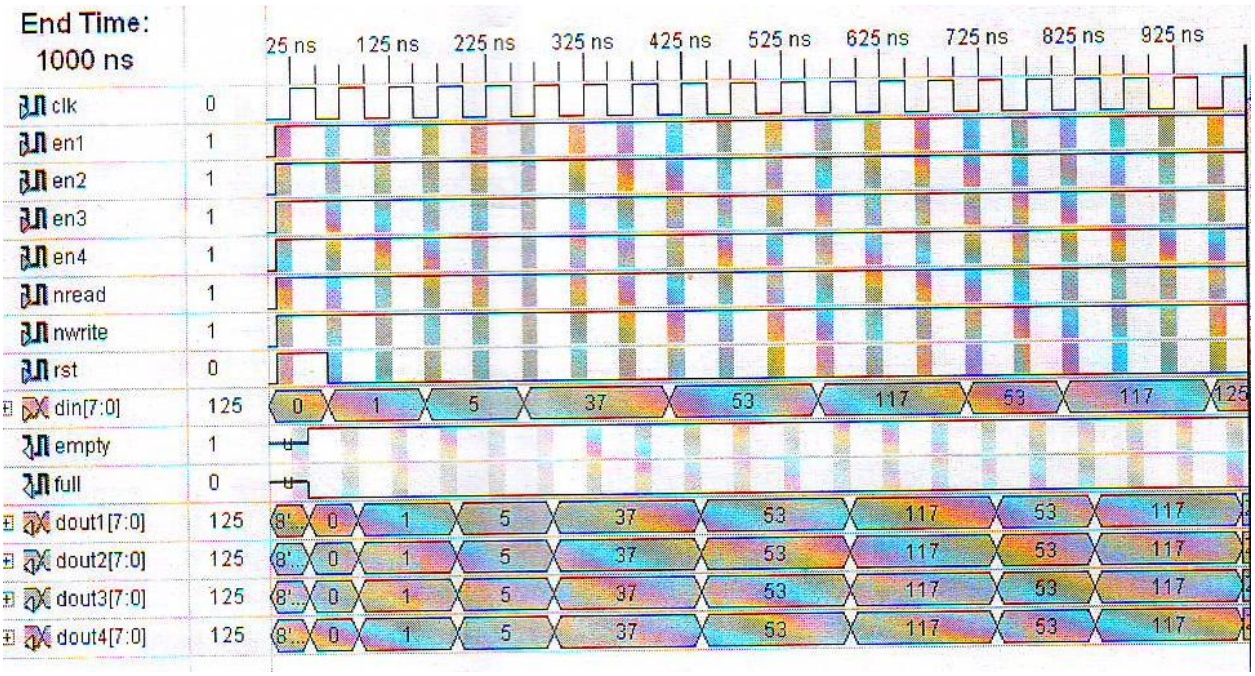

Figure 8 : Simulation results for same baud rates

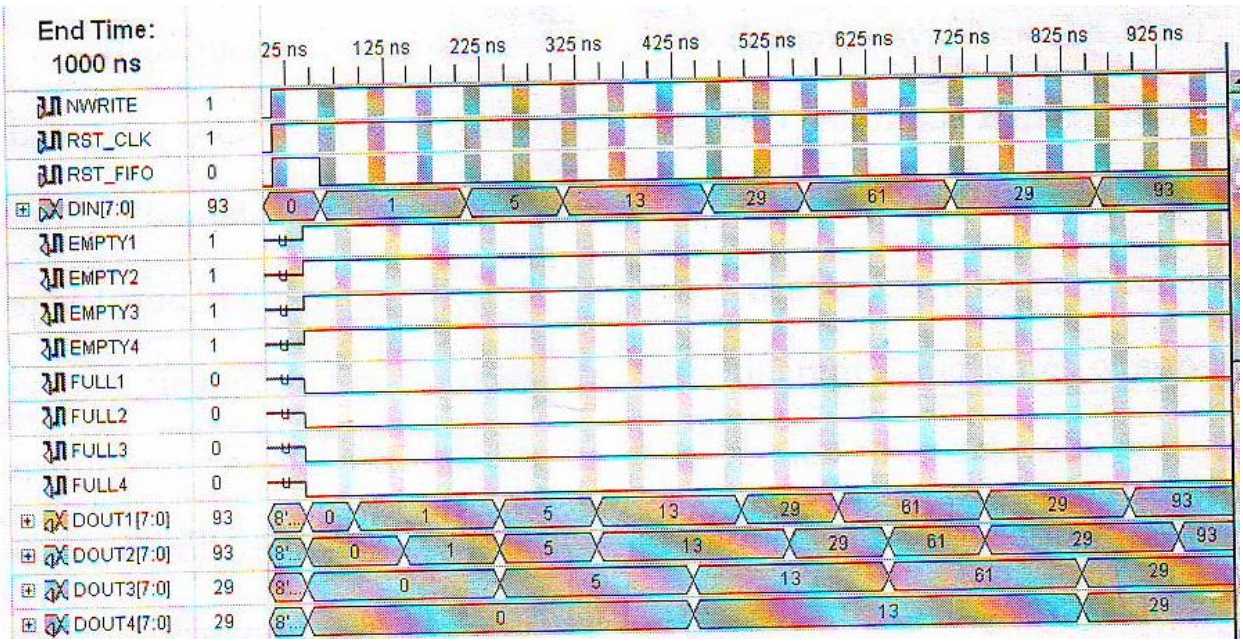

Figure 9 : Simulation results for different baud rates

IV. Conclusion

This paper introduces a method to design a method to design a asynchronous FIFO based on FPGA. And using asynchronous FIFO technique implements a multi channel UART controller within FPGA based on SRAM wit high speed and reliability .The controller can be used to implement communications in complex system with different baud rates of sub controllers. And it also can be used to reduce time delays between sub controllers of a complex control system to improve the synchronization of each sub controller. The controller is reconfigurable, so the controller"s fault is that it would be influenced by the radiation from surroundings and by short time pulses easily.

References Références Referencias

1. L.K.Hu and Q.CH wang,"UART –Based reliable Communication and performance analysis ",Computer Engineering Vol 32 No.10,May 2006,pp 5-21.
2. F.S.pan,F.ZHAO,J.xi and Y.Luo,"Implementation of Parallel Signal processing system based on FPGA and multi-DSP",Computer Engineering Vol 32,No.23,Dec 2006,pp247-249.
3. William stallings, "Data and Computer communications",McGraw-Hill publications.
4. Stephen Brown,Zvonko Vranesic"Fundamentals of Digital Logic with VHDL Design ".2nd edition, McGraw–Hill, Hardcover, Published July 2004.
5. Charles H.Roth "Digital system design using VHDL"2nd edition, Thomson publication.
6. Jayram bhaskar "A VHDL Primer "2nd edition Prentice Hall Publications.
7. Zainalabedin Nawabi "VHDL Analysis and Modeling of Digital systems" 2nd edition, McGraw-Hill, Hardcover, Published January 1998.
8. X.D.Wu and B.Dai, "Design of interface Between High speed A/D and DSP based on FIFO", Journal of Beijing Institute of petrochemical Technology Vol 14 No.12,June 2006,pp26-29.