Performance of DMA Mode UART IP Soft Core in Embedded Systems

S.Swetha, N.Vijaya, Lakshmi

Assistant professor's in Department of ECE, Mallareddy college of Engineering, Hyderabad, India

Abstract
In this paper presents performance of DMA mode UART IP soft core in embedded systems. In the past, UART IP hard core is less flexibility, low performance and more time taken by the CPU. So, the performance of embedded system is less. In this paper used UART IF soft core based on DMA is used. It has less occupancy time of CPU and improving the performance of the whole NIOSII system. Design requirements can be better met because of its high-performance, configurable parameters, portability besides high flexibility, practicality. It is design four sub modules and verify in a NIOSII embedded hardware system.

Index term:
NIOSII; UART; IP; DMA; AVALON bus

I. Introduction

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in the dsPIC33F/PIC24H device family. The UART is a full-duplex, asynchronous communication channel that communicates with peripheral devices and personal computers using protocols such as RS-232, RS-485, LIN and IrDA®. The module also supports the hardware flow control option with UxCTS and UxRTS pins and includes the IrDA encoder and decoder. It transmit 9600 to 38400 bps for transmitting data bit Whole process of serial transmission is based upon the principle of shift register[1]. There are two primary forms of serial transmission: Synchronous and Asynchronous.

Synchronous serial transmission requires that the sender and receiver share a clock with one another, or that the sender provide a strobe or other timing signal so that the receiver knows when to "read" the next bit of the data. Asynchronous serial communication has advantages of less transmission line, high reliability, and long transmission distance, therefore is widely used in data exchange between computer and peripherals. This Asynchronous serial communication is usually implemented by UART [2]. The entire UART IP soft core in DMA mode mainly includes the following 5 sub-modules: UART send controller, UART Receive controller, Register file with the Interface of Avalon-MM Slave [3], Master Read type DMA controller with the interface of Avalon-MM Master [4] and Master Write type DMA controller with the interface of Avalon-MM Master.

Manuscript received October 5, 2015
Manuscript revised October 20, 2015

It has poor flexibility, the small application, and the poor transportability; it's usually unable to meet the high requirements of the customer.

In this paper, UART IP soft core based on DMA mode can reduce elapsed time of CPU greatly in data transmission process so that the performance of NIOSII system can be improved and design requirement can be better met with less resources occupied, high speed, high flexibility and high transportability.

II. Desigh Of Digital Circuits in Verilog

In integrated circuit technology, the design of gate level is more time consuming so, UART IP core is used in VERILOG. VERILOG can be used to describe and simulate the operation of digital circuits ranging from few gate to more complex gates. VERILOG can be used for the behavioral level design implementation of a UART and it offers several advantages. These are the advantages of using VERILOG to implement UART:

1. VERILOG allows us to describe the function of the transmitter in a more behavioral manner rather than focus on its actual implementation at the gate level .
2. VERILOG makes the design implementation easier to read and understand.
3. It is easier to test the UART by the VERILOG simulation and find out if any discrepancy occurs.[2]

III. UART IP Soft Core in DMA

It has consumed time and high flexibility compare to UART IP hard core module. The block diagram of UART IP soft core in DMA as shown below.

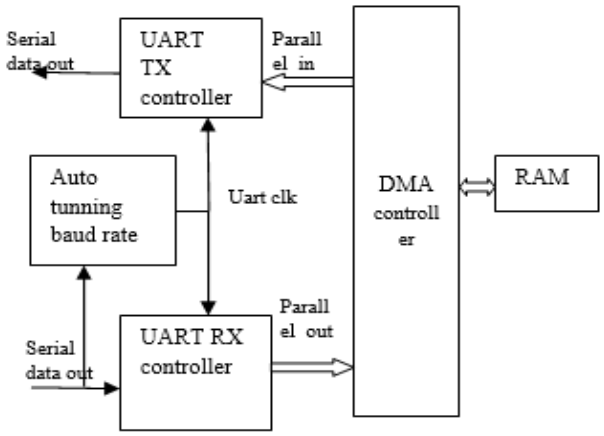

Fig.1. block diagram of UART IP soft core based DMA

Figure 1shows a block diagram of UART IP soft core with DMA which mainly consists of four sub modules
The detail design of each sub-module is as follows.

i) Auto-tuning baud rate generator

Baud rate generator is used to provide the reference time of sending and receiving data for transmitter module and receiver module. The serial data is in the form of a stream of '1's and '0's. '1' signifies 'high', and '0' signifies 'low'. Only repetitive measurements of the smallest occurring interval between data transitions can establish the incoming data baud rate. The method described here uses digital counters, edge detectors, comparators, and register storage of the time interval data from which the baud rate is easily computed.

ii) UART Transmitter

When a word is given to the UART for asynchronous transmissions, a bit called the "Start Bit" is added to the beginning of each word that is to be transmitted. The Start Bit is used to alert the receiver that a word of data is about to be sent, and to force the clock in the receiver into synchronization with the clock in the transmitter.
After the Start Bit, the individual bits of the word of data are sent, with the Least Significant Bit (LSB) being sent first. When the entire data word has been sent, the transmitter may add a Parity Bit that the transmitter generates. The Parity Bit may be used by the receiver to perform simple error checking. Then at least one Stop Bit is sent by the transmitter.

iii) UART Receiver

When the receiver has received all of the bits in the data word, it may check for the parity bits (both sender and receiver must agree on whether a parity bit is to be used), and then the receiver looks for a stop bit.

Regardless of whether the data was received correctly or not, the UART automatically discards the start, parity and stop bits. Then the receiver reassembles the serial bits into parallel data and loads into a data buffer. If the sender and receiver are configured identically, these bits are not passed to the host. If another word is ready for transmission, the start bit for the new word can be sent as soon as the stop bit for the previous word has been sent. Next section describes the inputs and output signals of receiver.

iv) DMA (Direct Memory Access)

The DMA is intended to be used in low power mode because it uses the same memory bus as the CPU and only one or the other can use the memory at the same time.A DMA controller is a device, usually peripheral to a computer's CPU, which is programmed to perform a sequence of data transfers on behalf of the CPU. A DMA controller can directly access memory and is used to transfer data from one memory location to another, or from an I/O device to memory and vice versa. A DMA controller manages several DMA channels, each of which can be programmed to perform a sequence of these DMA transfers. Devices, usually I/O peripherals, that acquire data that must be read (or devices that must output data and be written to) signal the DMA controller to perform a DMA transfer by asserting a hardware DMA request signal.
A DMA request signal for each channel is routed to the DMA controller. This signal is monitored and responded to in much the same way that a processor handles interrupts.When the DMA controller sees a DMA request, the DMA controller responds by performing one or many data transfers from that I/O device into system memory or vice versa. Channels must be enabled by the processor for the DMA controller to respond to DMA requests.

IV. REULTS

This section explains the Simulation Results and Synthesis Report of UART with DMA implementation.
Each module is coded in Verilog and simulated. The simulation results and the net list simulation are verified for each module by Model sim SE 10.0 and Xilinx 13.2

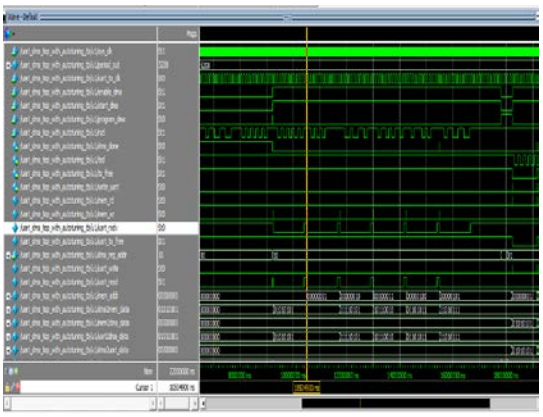

Figure 2: UART IP Soft core top level waveforms.

Figure 2 shows top module simulation where the received data 01010101 i.e. from uart2dma then dma2memory and then data is transmitted serially by the transmitter i.e. from mem2dma to dma2uart.

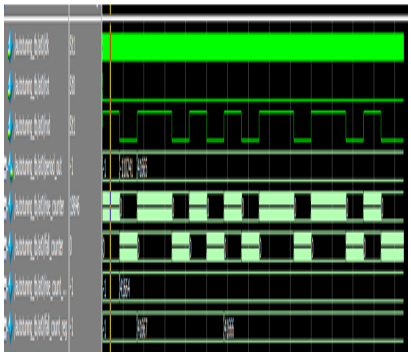

Figure 3: Auto tuning baud rate generator waveforms.

Figure 3 shows auto-tuning baud rate simulation result where the rise counter register is updated with count 41664 whenever fall edge detected and similarly fall counter register is updated the average of these two registers gives the baud divisor value here 41665 is for 1200bps baud rate.

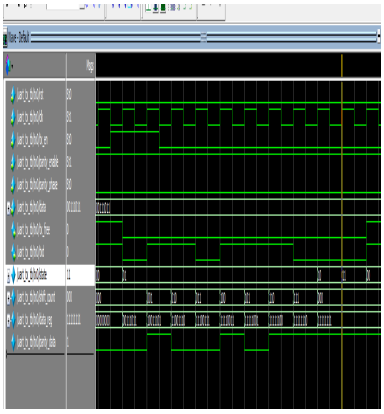

Figure 4: UART Transmitter waveforms

Figure 4 shows uart transmitter simulation. As this is designed using four states in idle state[00] data register holds zero, shift[01] gets parallel data only when start bit is low and starts shifting the data until the total 8bits are transmitted, parity state[10] transmits the parity bit and in last state[11]done includes stop bit high.

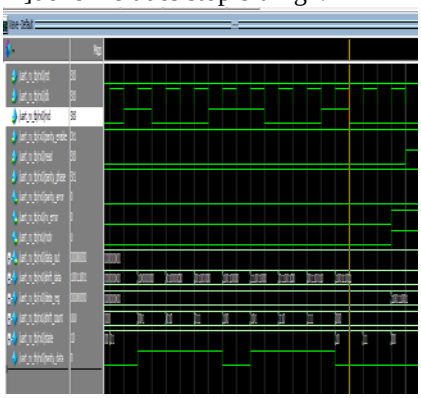

Figure 5: UART Receiver waveforms

Figure5shows uart receiver simulation waveforms. first it detects start bit of received serial data in idle[00] state then it starts shifting the data in shift state[01], checks for parity error in parity state[10]and last detects stop bit and loads the parallel data

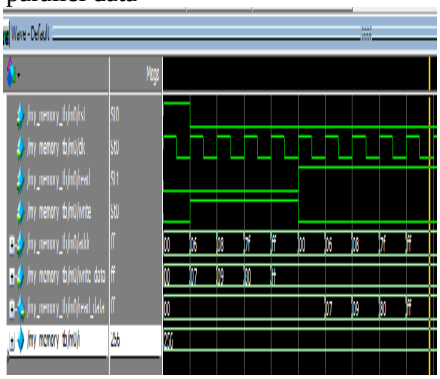

Figure 6: Memory waveforms

Figure 6 shows memory simulation results with specified memory locations read and write data.

Figure 7 shows DMA simulation waveforms where data transferred from source to destination .256 is the destination address i.e. transmitter gets data from mem2dma similarly 257 is the source address i.e. receiver uart2dma from which data is write into memory location 4 i.edma2mem.

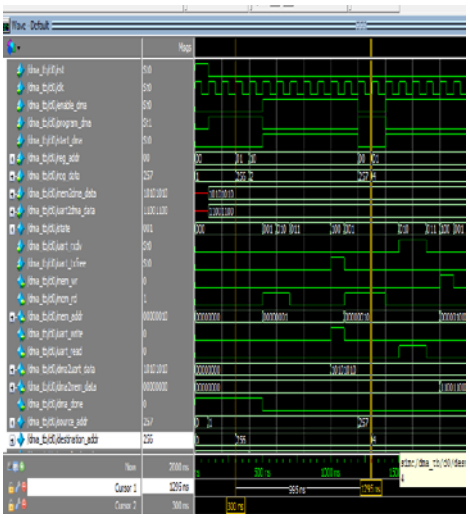

Figure 7: DMA waveforms

V. CONCLUSION

This paper describe the UART IP core based DMA has been proposed. This technique has high flexibility and less consume time of CPU that the performance of embedded system is high. The baud rate generator and auto-tuning approach in UART IP core based are simple and flexible, which can be applied to other application designs as a separate core.

REFERENCES

[1] Ananya Chakraborty, Surbhi, Sukanya Gupta, Swati Deshkar, Pradeep Kumar Jaisal,"Design of UART (Universal Asynchronous Receiver Transmitter) using VHDL", IJCST Vol. 3, Issue 1, Jan. - March 2012.

[2] FANG Yi-Yuan CHEN XUE- jun , "Design and simulation Of UART serial communication Module Based on VHDL",2011.

[3] Altera Corp, Avalon-MM Slave reference manual.

[4] Altera Corp, Avalon-MM Maste reference manual.

S. SWETHA is an Assistant professor at Malla Reddy College of Engineering, Hyderabad. She received her Bachelor degree in Electronics and communication Engineering from st.stanley college of engg and technology, hyderabad. M.Tech, Embedded & V.L.S.I System Design from osmania university,hyderabad.

N.VIJAYA LAKSHMI is an Assistant professor at Malla Reddy College of Engineering, Hyderabad. She received her Bachelor degree in Electronis and communication Engineering from QIS college of engg and technology, ongole, prakasam dist in AndhraPradesh. M.Tech, V.L.S.I System Design from Gitam university ,vizag in Andhra pradesh