Design, Development and Verification of Synchronous Serial Port (S-PORT)

Nagaraju.Sangepu
Associate Professor, CSE
Kakatiya Institute of Technology &Sciences
Warangal, India.
nag.sangepu@gmail.com

Uma Mithra.Bayya
VLSI & ES
Kakatiya Institute of Technology & Sciences
Warangal, India.
mithra4c3@gmail.com

***Abstract*— The objective of this project is to design S-PORT Core using Verilog and verify the code using system verilog. Serial Synchronous Ports or S-PORT's is an interface that facilitates the transmission of synchronous serial data. S-PORT is a synchronous serial data link that operates in full duplex mode. It can provide a direct interconnection between processors in a multiprocessor system. They support a variety of serial data communication protocols and they can also provide a direct interconnection between processors in a multiprocessor system. Serial Port data can be automatically transferred to and from ON-Chip Memory using DMA block transfers. Bidirectional functions provide flexibility for serial Communication. Serial Communication Ports can operate at half the full clock rate at the processor, at a maximum data rate of n/2 Mbps, where n-equals the processor core clock frequency.**

Keywords-S-PORT Interface; serial; System Verilog; Verilog HDL.

I. INTRODUCTION

Serial data transmission is widely used in communications over long distances. Parallel communication requires many wires to be laid between the two communicating points. Hence, usually data is converted to serial format and sent over fewer numbers of wires to the destination. To interface a microcomputer with serial data lines, the data must be converted to and from serial form. A parallel-in-serial-out shift register and a serial-in-parallel-out shift register can be used to do this. Also needed of some cases of serial data transfer data is hand-shaking circuitry to make sure that a transmitter does not send data faster than it can be read in by the receiving system. A device which can be programmed to do synchronous communication, is often called as synchronous serial peripheral port or S-PORT.

Many microprocessor devices have a built in S-PORT and it is one of the commonly used serial interface peripherals. As a peripheral device of a microcomputer system, the S-PORT receives parallel data from the C.P.U and performs parallel to serial conversion at the transmitter end and serial to parallel conversion at the receiver end. Serial Communication Ports can operate at half the full clock rate at the processor, at a maximum data rate of n/2 Mbps, where n-equals the processor core clock frequency.

S-PORT can transfer a frame of data with three or thirty two bits per transmission.

S-PORT is mostly used in system on chip (SOC) which helps a multiprocessor communication within a single PCB. This happens through the synchronous serial peripheral port where in one serial peripheral port communicates with another serial peripheral port of another processor through a suitable communication channel. At present dual core processors are mainly used in any system due to its various advantages and applications in real time. Due to the changes like reduction of size and various high speed applications, this S-port is a basic module which changes the system's behavior to emerge into a high speed processor capability system.

All ADSP-218x family processors contain two serial ports, S-PORT0 and S-PORT1. These serial ports have some similarities and some differences. This paper provides a detailed description of the S-PORTs and explains the differences between the two.

II. BASIC DESCRIPTION OF S-PORT

A. Synchronous Serial Port (SSP)

The Synchronous Serial Port (SSP) is a serial interface useful for Communicating with other peripherals or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. Each S-PORT has a five-pin interface: Serial clock(SCLK),Receive frame synchronization(RFS),Transmit frame synchronization(TFS),Serial data receive(DR),Serial data transmit(DT). All ADSP-218x family processors contain two serial ports, S-PORT0 and S-PORT1 as shown in fig [1].

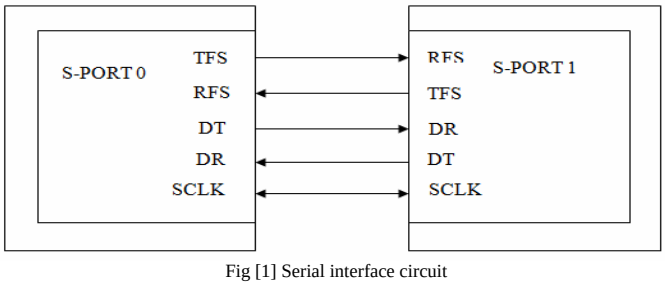

Fig [1] Serial interface circuit

A S-PORT receives serial data on its DR input and transmits serial data on its DT output. It can receive and

transmit simultaneously for full duplex operation. The data bits are synchronous to the serial clock SCLK, which is an output if the processor generates this clock or an input if the clock is generated externally. Frame synchronization signals RFS and TFS are used to indicate the start of a serial data word or stream of serial words. Fig [2], shows a simplified block diagram of a single S-PORT

B. *Block Diagram of S-PORT*

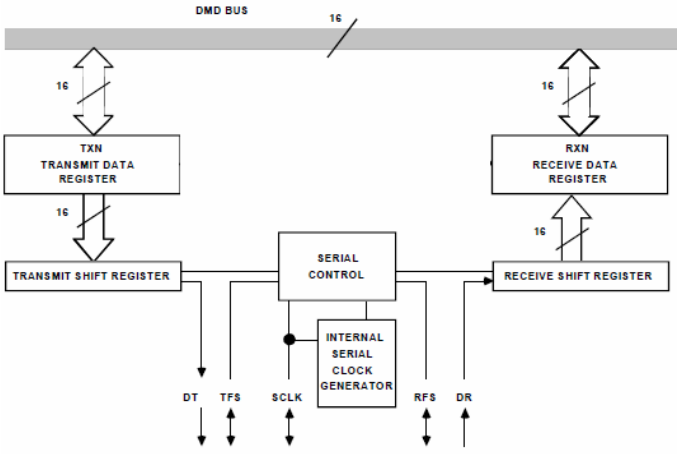

Fig [2] Serial Port Block Diagram

Data to be transmitted is written from an internal processor register to the S-PORT's TX register via the DMD bus. The bits in the Shift register are shifted out on the S-PORT's DT pin, MSB first, synchronous to the serial clock. The receive portion of the S-PORT accepts data from the DR pin, synchronous to the serial clock. When an entire word is received, then it is automatically transferred to the S-PORT's RX register, where it is available to the processor.

C. *Operation of S-PORT*

Writing to a S-PORT's TX register readies the S-PORT for transmission: The TFS signal initiates the transmission of serial data. Once transmission has begun, each value written to the TX register is transferred to the internal transmit shift register and subsequently the bits are sent, MSB first. Each bit is shifted out on the rising edge of SCLK. After the first bit (MSB) of a word has been transferred, the S-PORT generates the transmit interrupt. The TX register is now available for the next data word, even though the transmission of the first word is ongoing.

In the receiving section, bits accumulate as they are received in an internal receive register. When a complete word has been received, it is written to the RX register and the receive interrupt for that S-PORT is generated.

D. *S-PORT Enable*

S-PORTs are enabled through bits in the System Control register, as shown in Fig [3].This register is mapped to Data Memory address 0x3FFF. Bit 12 enables S-PORT0 if it is a 1, and bit 11 enables S-PORT1 if it is a 1. Both of these bits are cleared at reset, disabling both SPORTs.

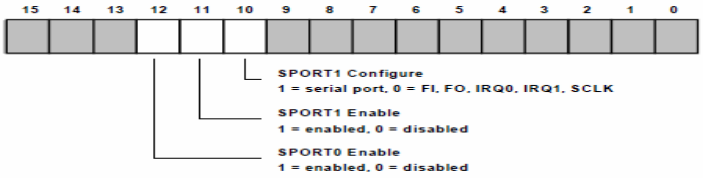

Fig [3] SPORT Enables in the System Control Register

III. DESIGN PRINCIPLES

Verilog HDL is a hardware description language, it can be used for different levels of logic design, can be used for digital system logic simulation, timing analysis and logic synthesis, a wide range of applications. In this paper, a S-PORT Interface Module is designed using verilog, to achieve a multiprocessor communication within a single PCB. This happens through the synchronous serial peripheral port where in one serial peripheral port communicates with another serial peripheral port of another processor through a suitable communication channel.

A)Module design:

The modules which are to be individually coded and to be Instantiated are

1) Clock generator
2) FIFO
3) Transmitter-Receiver logic
4) C.P.U interface

B)Characteristics Of S-PORT

Many of the S-PORT characteristics are configurable to allow flexibility in serial communication.

- *Bidirectional*—Each S-PORT has independent transmit and receive sections.
- *Double-buffered*—Each S-PORT section (both receive and transmit) has a data register for transferring data words to and from other parts of the processor and a register for shifting data in or out. The double-buffering provides additional time to service the S-PORT.
- *Clocking*—Each S-PORT can use an external serial clock or generate its own in a wide range of frequencies down to 0 Hz.

$$\text{SCLK frequency} = \frac{\text{CLKOUT frequency}}{2\times(\text{SCLKDIV}+1)}$$

SCLKDIV	SCLK Frequency
30719	1200 Hz
3839	9600 Hz
575	64 kHz
23	1.536 MHz
17	2.048 MHz
5	6.144 MHz

Table [1] Common Serial Clock Frequencies (Internally Generated)

- *Word length*—Each S-PORT supports serial data word lengths from three to sixteen bits.
- *Framing*—Each S-PORT section (receive and transmit) can operate with or without frame synchronization signals. For each data word with internally-generated or externally-generated frame signals, with active high or active low frame signals with either of two pulse widths and frame signal timing.
- *Auto buffering with single-cycle overhead*—Using the DAGs, each S-PORT can automatically receive and/or transmit an entire circular buffer of data with an overhead of only one cycle per data word. Transfers between the S-PORT and the circular buffer are automatic in this mode and do not require additional
- *Interrupts*—Each S-PORT section (receive and transmit) generates an interrupt upon completing a data word transfer, or after transferring an entire buffer if auto buffering is used
- *Multichannel capability*—S-PORT0 can receive and transmit data selectively from channels of a serial bit stream that is time-division multiplexed into 24 or 32 channels. This is especially useful for T1 interfaces or as a network communication scheme for multiple processors.

IV. SIMULATION AND VERIFICATION

Using Verilog language description design the S-PORT interface circuit, synthesis with ISE, and then use Questasim to simulate. Xilinx synthesis technology (XST) tool is selected for synthesis. Through this S-PORT functions as a transmitter as well as a receiver.

The verilog model is simulated by Questasim software for the purpose of the testing verilog model. Each and every individual block is coded in verilog and instantiation is done then the simulation results are observed. A clock generator produces a frequency of about 12Mhz. This frequency is reduced to a 6Mhz signal in order to support S-PORT design. The simulation waveforms of the corresponding modules are shown below [4]-[8].

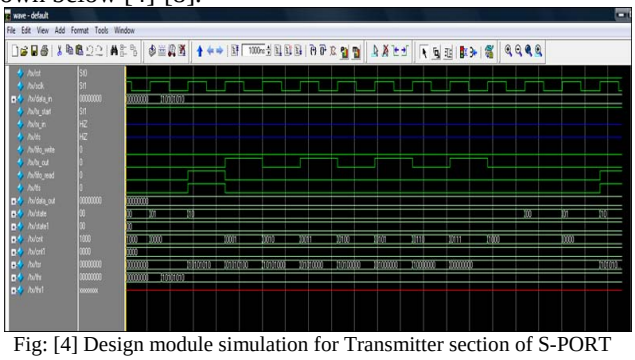

Fig: [4] Design module simulation for Transmitter section of S-PORT

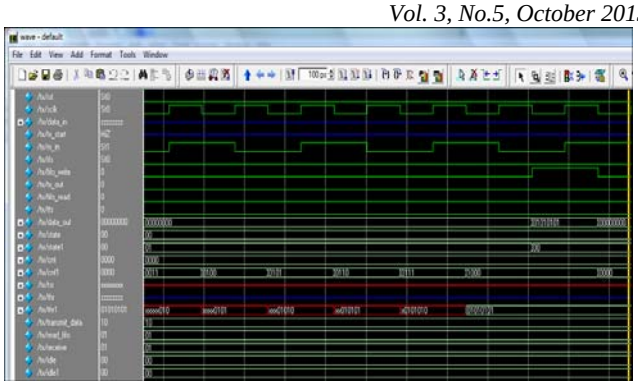

Fig: [5] Design module simulation for Receiver section of S-PORT

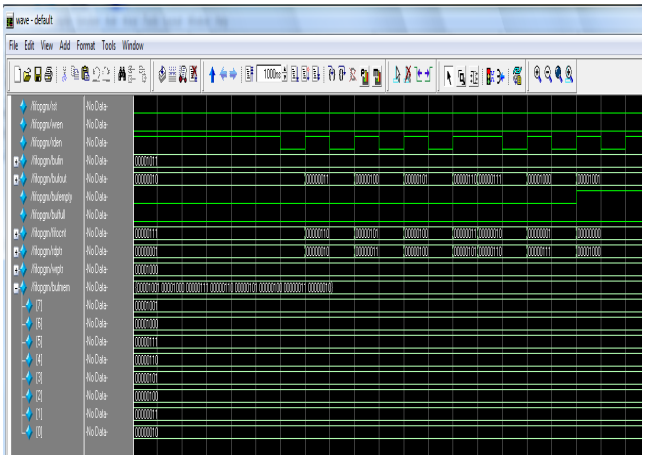

Fig: [6] Design module simulation for FIFO_READ

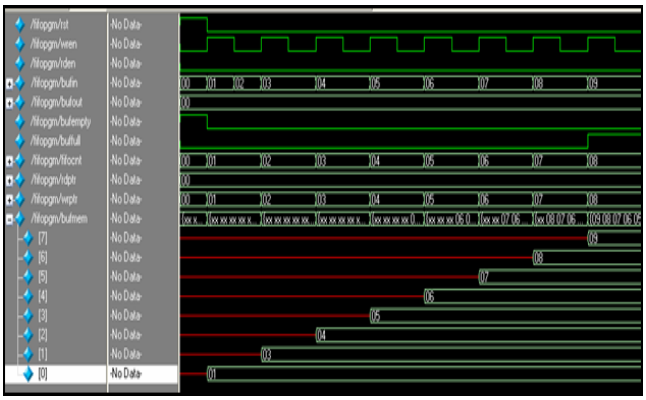

Fig: [7] Design module simulation for FIFO_WRITE

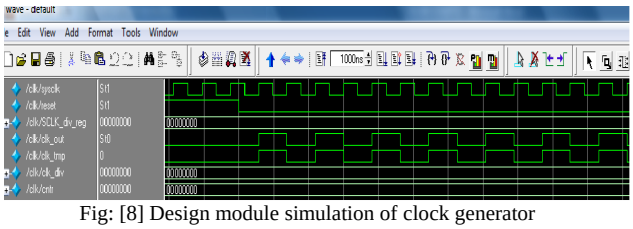

Fig: [8] Design module simulation of clock generator

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	632	69120	0%
Number of Slice LUTs	790	69120	1%
Number of fully used LUT-FF pairs	447	975	45%
Number of bonded IOBs	58	640	9%
Number of BUFG/BUFGCTRLs	5	32	15%

Fig:[9] Device Utilization Summary Report

Verification is also the major work of this paper because as we are using many modules and having many inputs we are going to use System verilog for verifying the design. Here the Design is considered as design under test (DUT). Through verification we can send the stimulus according to our wish. Here we can even constraint the unwanted stimulus. Components of Verification Environment are as follows: RTL module, Interface module to connect DUT with Test Bench, Transaction, Generator, Driver, Receiver, Scoreboard, TOP level module which encapsulates the above all with RTL and Test Cases.

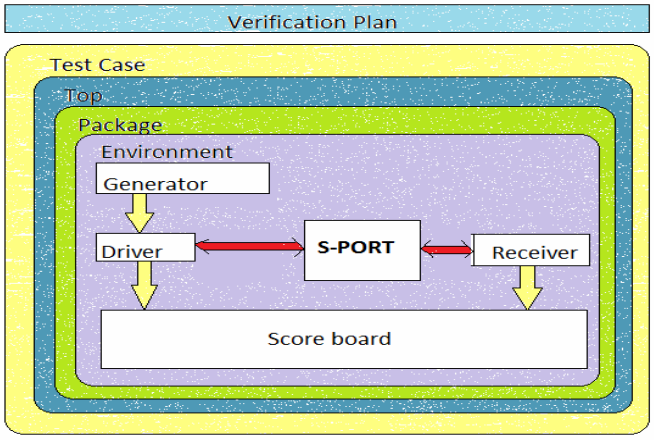

Fig [10] S-PORT-Top Test bench overall structure

V. CONCLUSION

In this, paper we have designed the Synchronous Serial Ports which are mainly used in multiprocessor communication where the data transmission can takes place at a high speed within the system (i.e.. For short distances) and accuracy is also achieved. So this Port is designed for each module in verilog code and functional verification is also done.

To interface a microcomputer with serial data lines, the data must be converted to and from serial format hence Serial communication interface is used which can perform this function. These interface devices can be operated in the synchronous mode. In order to facilitate the data exchange between these devices there is a need for an effective interface. S-PORT is one such serial communication interface device in the synchronous mode that helps in achieving error free data transmission and can be further extended for more data bits.

ACKNOWLEDGMENT

Sangepu NagaRaju, attained his B.Tech in Computer Science Engineering from Amaravati University ,Andhra Pradesh, India, in 1987 and M.Tech in Computer Science & Engineering from JNTU, Hyderabad, Andhra Pradesh, India in 2007. He is pursuing his Ph.D in Semantic Web, JNTU, Hyderabad, Andhra Pradesh ,India .He is now working as Associate Professor in the department of Computer Science Engineering in Kakatiya Institute of Technology and Science, Warangal, Andhra Pradesh, India since August 1998. His research interests include Data Mining, Genetic Algorithm, Evolutionary Algorithms, Semantic Web Mining.

Uma Mithra Bayya was born in Warangal, India in 1990. She received B.Tech degree in Electronics and Communication Engineering from JNTU, Hyderabad in 2011. She is pursuing her Masters degree in VLSI & ES in Kakatiya Institute of Technology and Science, Warangal, India. Her research interests include Communication Systems and Digital Signal Processing.

REFERENCES

[1] ADSP-218x DSP hardware reference, first edition, February 2001.
[2] www.xilinx.com/support/documentation
[3] www.xilinx.com/support/documentation/sw_manual/.../xst.pdf
[4] www.analog.com/processors and DSP/ Blackfin processors/
[5] www.xilinx.com/products/silicon-devices/fpga/virtex-5/index.htm
[6] Forums.xilinx.com silicon devices virtex R family FPGAs
[7] Basic VLSI design by Douglas A.pucknell ESHRAGHIAN, KAMRAN, third edition.
[8] National Instruments Serial Quick Reference Guide, February 2007.
[9] Jiang Lidong. VHDL language program design and application.Beijing: Beijing University of Posts and Telecommunications Press, 2004.
[10] Wang Shiyi. Digital signal processing. Beijing: Beijing Institute of Technology Press, 1997.