International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Design and Performance Analysis of Serial Peripheral Interface

Gaurav Anand[1], Shelly Gahlawat[2]

[1, 2]Department of Electronics, Amity University, Noida, India

Abstract: ***Today Serial Communication has become effective in terms of cost and synchronization. Serial Peripheral Interface (SPI) is a type of serial communication protocol which operates in full duplex mode and connects various devices and microcontrollers. In this paper we implement Serial Peripheral Interface using VERILOG AND XILINX ISE 13.1 and verify the design and obtain the coverage report using QUESTA SIM. Through this paper we analyse the Power, Floor Planning and Routing and customise them.***

Keywords: SPI, Comunication, XILINX ISE 13.1, XTAL, link

1. Introduction

Today apart from design and testing there is a need to know the power optimization of the device. Any design needs to be tested and verified. The need of verification along with testing environment creates a better protocol and a good intellectual property (IP). Serial peripheral interface is a serial interface use for serial communication. It always operates in full duplex mode and is used for interfacing with various devices and microcontrollers. Serial peripheral interface is implemented using Verilog and executed in Xilinx 13.1 and verified using Questa Sim. Apart from design and verification plan ahead is done which highlights floor planning and routing and power analysis.

2. Serial Peripheral Principles

Serial peripheral is an interface which uses minimal hardware and pins to transfer the bits per clock cycle. The design of the interface is done using the Hardware Descriptive Language (HDL) that has an advantage of concurrency than other programming language.SPI has two modes i.e master and slave. It has normally a single master and multiple slaves. Normally the master has four pins which are Master in Slave out (MISO), Serial Clock (SCK), Master out Slave in (MOSI) and chip select (SS). The chip select is normally a low signal while selecting a Slave device. For multiple slave devices the entire slave device has different chip select signal.

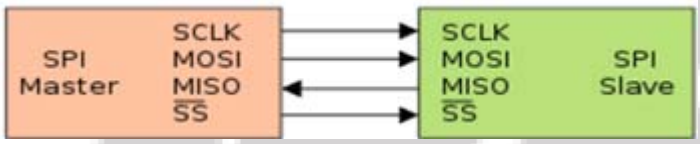


Figure 1: SPI schematic

SPI always operates in full-duplex mode. When the master transmits the data using MOSI line the slave at the same time sends the data to the master using MISO line.

3. Design Methodology

The SPI has been implemented using Hardware Descriptive Language (HDL). HDL are Verilog, VHDL and many.HDL has advantage of concurrency.SPI has been divide into different blocks which are XTAL, Clock Logic, Transmitter and Receiver. Each and every block has been executed using VHDL by writing the RTL in XILINX ISE 13.1 and the RTL schematic are generated.

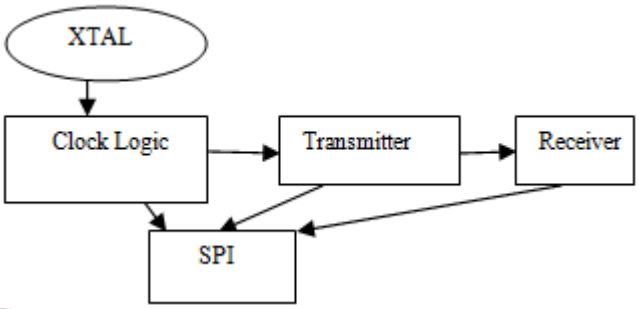
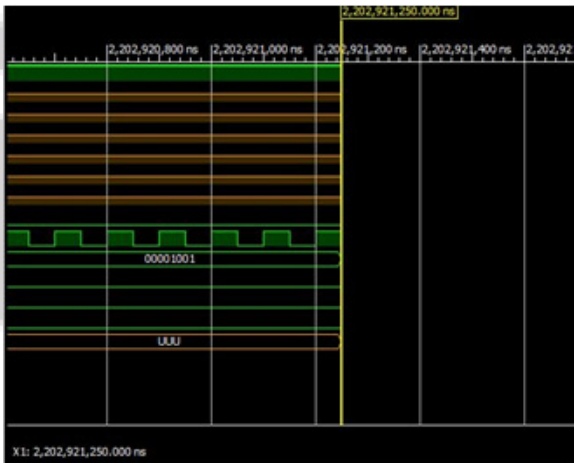


Figure 2:.Flowchart of the SPI design

A) XTAL Block-

The SPI operates at the frequency less than the maximum frequency of the slave. Thus for the stable operation the frequency is divided. There are different frequencies which are generated from divide by 2 to divide by 64.This frequencies are generated by selecting the SPR0 and SPR1. The clock frequency is changed by dividing the clock input signal.



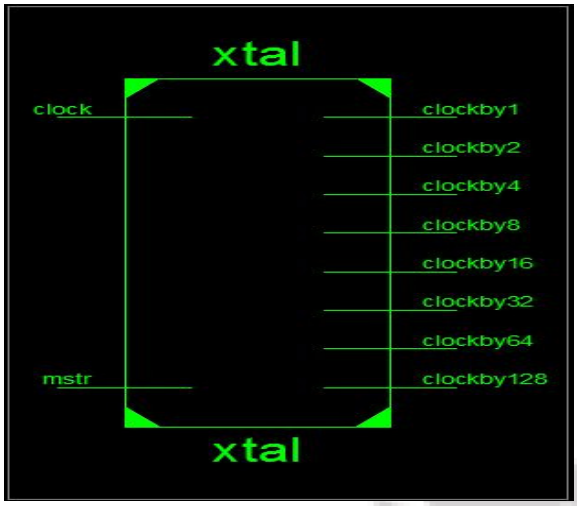
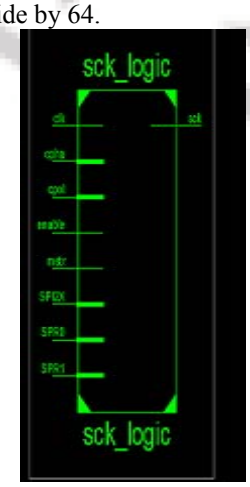


Figure 3: The schematic and waveform of XTAL

B) Clock Logic

The Clock Logic selects the frequencies generated by the XTAL block .The frequencies are selected using the SPI2X, SPR1, SPR0 bits. Clock Logic selects the frequencies from divide by 2 to divide by 64.



C) Transmitter Register

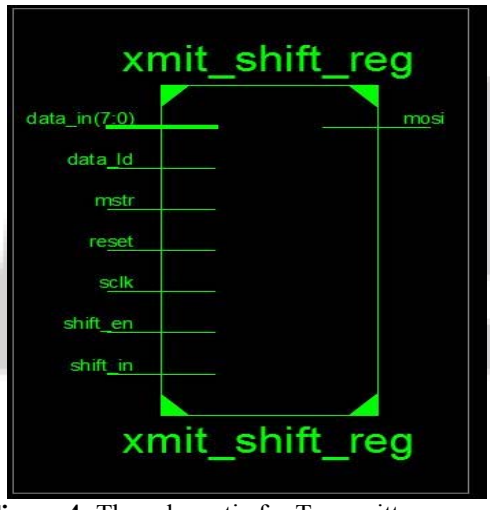


Figure 4: The schematic for Transmitter

4. Results

The implementation of SPI AND verification is done using the HDL and System Verilog. The results have shown the power analysis for the implementation of the SPI. The net list shows the interconnection between the devices.

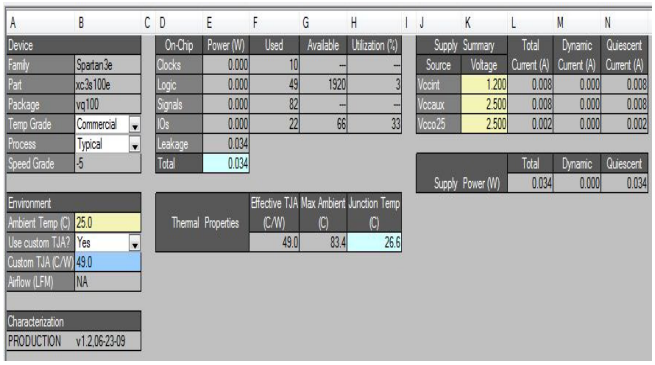


Figure 5: Power analysis of SPI

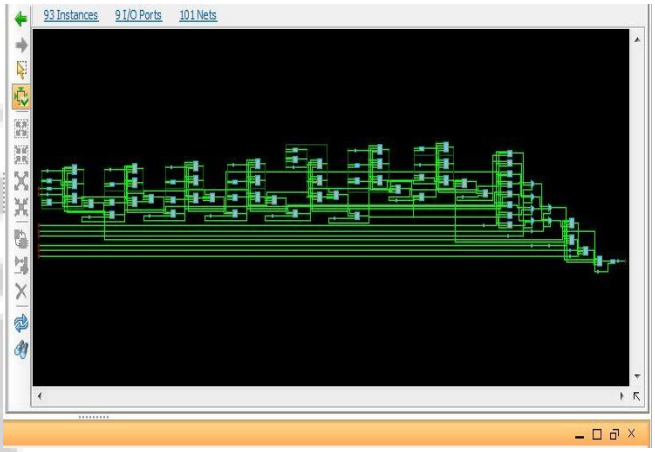


Figure 6: Netlist arrangement of SPI

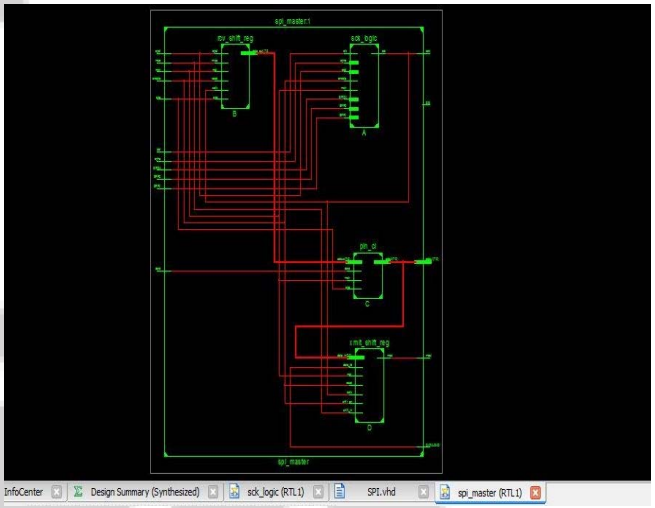


Figure 7: The interconnection of the components in SPI

S.NO.	Bin Name	Required	Covered	Percentage Coverage
1	CPOL_0-CPHA_0	1	2	100%
2	CPOL_0-CPHA_1	1	1	100%
3	CPOL_1-CPHA_0	1	1	100%
4	CPOL_1-CPHA_1	1	2	100%
5	Master_Data_below10	1	2	100%
6	Master_Data_above10	2	4	100%
7	Slave_Data_below10	1	1	100%
6	Slave_Data_above10	2	5	100%

Figure 8: The coverage report for verification

5. Conclusion

This paper focuses on the performance analysis of Serial Peripheral Interface along with the design work. Two separate works has been done and they are Design and Verification of Serial Peripheral Interface. Design part involves around the specifications and the code execution in the VHDL .The language is used to write the RTL for the interface. Serial Peripheral Interface is divided into five blocks and the RTL code is written for each of the above blocks. The authentic work is done using XILINX ISE 13.1 which gives the schematic of each of the block and thus validating the specifications. The output waveforms are in analogues to the blocks.

After the design work there is main focus on verification. Verification is done to check the functionality of the design and verifies whether the design is bugless. The verification is done by Hardware verification Language which is System Verilog. The verification is done by using assertions and setting the number of bins. Bins are similar to data types. The verification environment consists of coverage report with all the coverage report.

6. Acknowledgement

We hereby thanks everyone associated with the work and our professors who were always with us throughout this work.

References

[1] Motorola MMC 2001 Reference Manual.
[2] Ananthula Srinivas, M.K.Kumar and Jugal Kishore Bhandari," Design and Verification of Serial Peripheral Interface", International Journal of Engineering Development and Research, ISSN: 2321-9939.
[3] A.K. Oudjida et al, "FPGA Implementation of I2C and SPI Protocols: A Comparative Study". Proceedings of the 16th edition of the IEEE International Conference on Electronics Circuits and Systems ICECS, pp. 507 -510, ISBN: 978-1-4244-5091-6, December 13-16 2009, Yasmine Hammamet, Tunisia.
[4] Motorola Inc., "SPI Block Guide V03.06," February 2003.