

Performance Analysis of Serial Peripheral Interface on FPGA

Sukhwinder Kaur

Student (M.Tech), Deptt. Of ECE
SUS College of Engg. & Technology
Tangori, Mohali (Punjab), India

Abhishek Godara

Asst. Professor, Deptt. Of ECE
SUS college of Engg. & Technology
Tangori, Mohali (Punjab), India

Amita Choudhary

Asst. Professor, Deptt. Of ECE
SUS college of Engg. & Technology
Tangori, Mohali (Punjab), India

Abstract— Serial Communication is the best solution when data transfer is required between ICs on a board or between IC subsystems connected by cables. Serial Peripheral Interface (SPI) and Inter-Integrated Circuit (I2C) bus are the two popular serial communication protocols for inter-chip and intra-chip data transfers. This paper discusses and compares the implementation of SPI (master-slave) configuration on Lattice and Xilinx FPGA families. For a comparative study, two configurations ; one with gated clock and other without gated clock , are implemented on FPGA and parameters like frequency, power, area and delay which affects the performance, are compared. Modelsim Altera Quartus II is used for test bench verification. Gated clock architecture has shown significant improvement in dynamic power dissipation and delay as compared to non gated architecture.

Keywords – SPI, I2C, FPGA, Power, delay.

I. INTRODUCTION

There are two ways for transferring data for inter-chip and intra-chip communication applications. One way is to transfer the data (8-bit or more) in one clock cycle. This data transfer is known as parallel data transfer. ISA, PCI, Printer are the examples of parallel data communication. Another technique is to transfer the data, one bit at a time i.e. 8-bit data will take 8 clock cycles to get transferred. This is known as serial data transfer. SPI, I2C, Ethernet, USB, RS-232 are the examples of serial data communication. However, serial data transfer is more advantageous as compared to parallel data transfer because of simple wiring and less interaction between conductors of serial cables [1] [2].

SPI and I2C are two most popular buses used for inter-chip and intra-chip data communication. Both SPI and I2C provide good support for communication with low-speed devices. But, SPI is better suited for single master single slave communication and for applications in which devices transfer data streams. Throughput of SPI is more than I2C. I2C is better suited for multi-master “register” access applications. SPI is used to communicate with variety of peripherals like sensors (mouse sensors), memory devices, real time clocks etc [3].

II. SERIAL PERIPHERAL INTERFACE

Serial Peripheral Interface (SPI) was originally designed by Motorola for communication with peripheral devices. SPI is a 3+N wire full duplex communication protocol where N is the number of devices connected to single master. SPI supports 8-bit and 16-bit data transfers. SPI is a single master bus in

which one master controls N number of slaves. SPI has a simple hardware implementation and uses shift registers to clock data in and out. For a single master and single slave SPI configuration 4 wires are used i.e. MISO, MOSI, SCLK and SS [4] [5]. The Fig1 shows a typical SPI configuration having four pins; MOSI, MISO, SS and SCLK. Both SPI master and SPI slave uses shift registers for transferring the data. Data is transferred either from MSB side or from LSB side. MISO (Master In Slave Out) and MOSI (Master Out slave In) are two pins on which data transfer takes place.

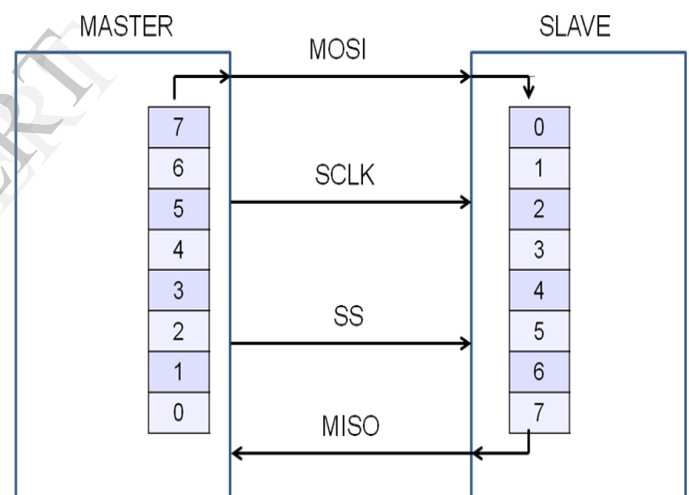


Figure 1. SPI communication with 8-bit shift registers

SCLK and SS (active low) are the control signals. SCLK is the SPI clock on which data transfer take place and SS is the slave select signal for selecting a slave for data transfer. There are two parameters associated with the SPI clock; CPOL (Clock Polarity) and CPHA (Clock Phase). These parameters will decide the polarity of the clock and edge on which data transfer takes place i.e. if value of CPOL is logic ‘0’ this means that the idle state of SCLK is low and if CPHA is logic ‘0’ then sampling of data occurs at odd edges (rising edges). So, there are total 4 modes of operation on the basis of CPOL and CPHA. Fig. 2 shows the concept of CPHA and CPOL.

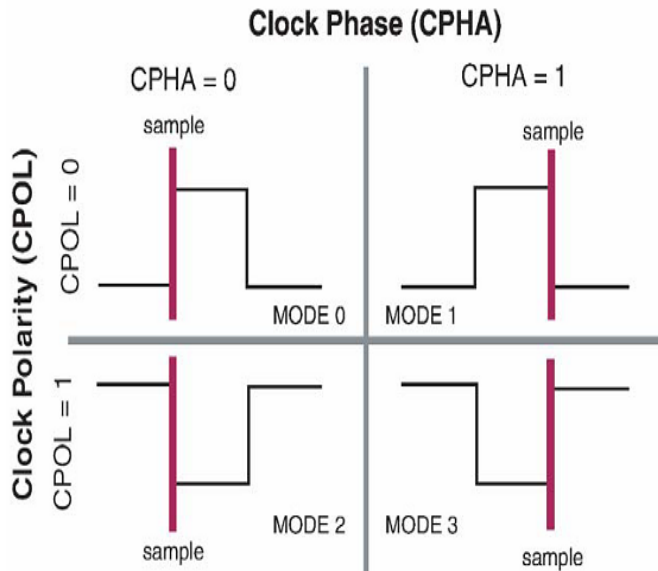


Figure 2. Different modes of SPI.

At CPOL=0 the base value of the clock is zero. For CPHA=0, data is captured on the clock's rising edge (low → high transition) and data is propagated on a falling edge (high → low clock transition). For CPHA=1, data are captured on the clock's falling edge and data is propagated on a rising edge. At CPOL=1 the base value of the clock is one (inversion of CPOL=0). For CPHA=0, data are captured on clock's falling edge and data is propagated on a rising edge. For CPHA=1, data are captured on clock's rising edge and data is propagated on a falling edge [6].

Baud rate register is used to specify the baud rate. Baud rate generation consists of a series of divider stages. Six bits in the SPI Baud Rate register (SPPR2, SPPR1, SPPR0, SPR2, SPR1, and SPR0) determine the divisor to the SPI module clock which results in the SPI baud rate. When all bits are clear (the default condition), the SPI module clock is divided by 2. When the selection bits (SPR2–SPR0) are “001” and the pre-selection bits (SPPR2–SPPR0) are “000”, the module clock divisor becomes 4. When the selection bits are “010”, the module clock divisor becomes 8 similarly 16, 32 values of clock divisor can be obtained [6].

The baud rate divisor equation is as follows:

$$\text{Baud Rate Divisor} = (\text{SPPR} + 1) \cdot 2^{(\text{SPR}+1)}$$

The baud rate can be calculated by following equation:

$$\text{Baud Rate} = \text{Bus Clock} / \text{Baud Rate Divisor}$$

III. DYNAMIC POWER REDUCTION: RTL CLOCK GATING

Power is a main parameter to take under consideration when better performance is the main goal to achieve. There are two types of power; static and dynamic which sums up to form total power. Clock power is the major component of chip dynamic power because it is fed to most of the IP cores in processors. This power can contribute up-to 50% of total

power. Dynamic power is directly proportional to frequency, power supply and load capacitance.

RTL clock gating technique is an effective technique for dynamic power reduction. Another advantage is that this technique reduces the routing burden and area to some extent, for example if there are 6 D-Flip Flops (Dffl) with common load signal, these flip-flops can be replaced with D flip-flops and a clock gating circuit. This results in reduction in routing efforts for the load signal to all the FFs [7].

Combinational clock gating, system level clock gating and sequential clock gating are the three types of clock gating techniques that are used. The first technique reduces the power by disabling the clock on registers when the output is not changing. This technique is very easy to implement. The second technique suspends the clock for entire design, disabling all effective functionality. The last technique is multi-cycle optimization technique with RTL modifications. It is very complex technique and difficult to implement [8][9][10].

IV. PROPOSED SYSTEM MODEL

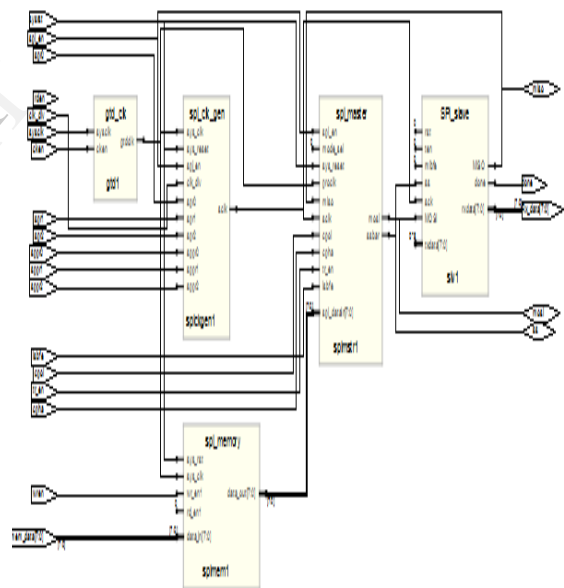


Figure 3. Gated clock SPI architecture

Fig.3 shows the RTL view of the gated clock architecture which consists of gated clock module, SPI memory, SPI clock generator, SPI master-SPI-slave. The system clock is gated by the gated clock module. When clock enable (ck_en) signal goes high the clock becomes active otherwise clock remain suspended. The role of SPI memory is to provide the data to the SPI master to transfer it to the corresponding slave. SPI clock generator generates the SCLK signal over which the data is to be transferred. The SCLK signal depends on the values of CPOL, CPHA and Baud rate register.

V. RESULTS AND SIMULATIONS

Fig. 4 and Fig.5 shows the test bench simulations of SPI master and SPI-slave modules in Modelsim Altera simulator.

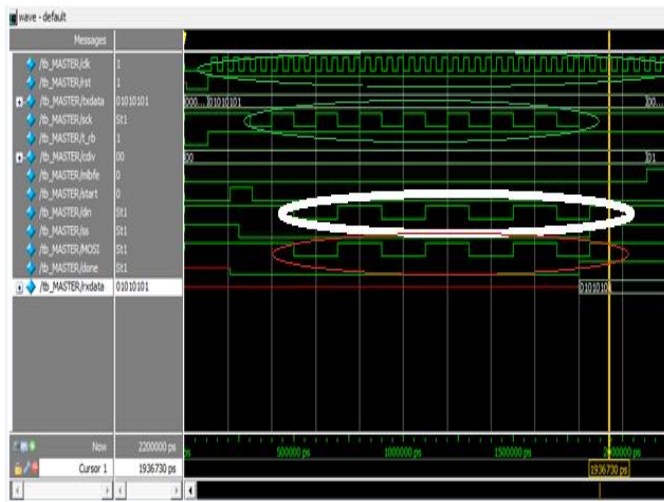


Figure 4. SPI master test bench waveform.

In Fig.4 the white marking region is showing the data in to SPI Tx register i.e. 01010101 and red marking region is showing the data transmission on MOSI. Green marking region is showing the SCLK which is div by 4 of the system clock (clk).

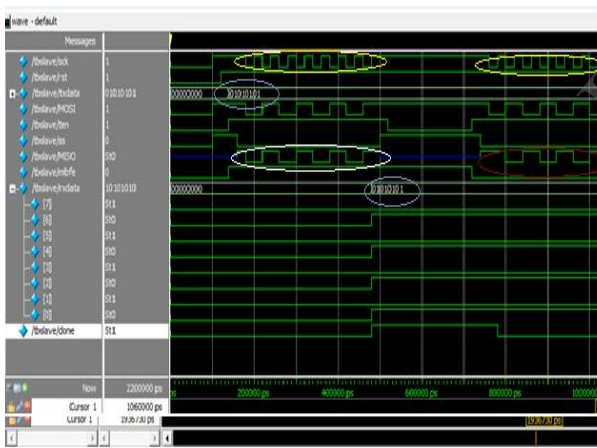


Figure 5. SPI slave test bench waveform.

In Fig.5 yellow marking region is showing the SCLK same as was in SPI master. White marking region is showing the data transmission on MISO. Blue marking region is showing the receive data (rxdata) from MISO i.e 01010101.

Fig.6 shows the hierarchal view of non-gated SPI on Lattice XP2 FPGA. As shown, there are four modules; SPI master, SPI slave, SPI memory and SPI clock generator.

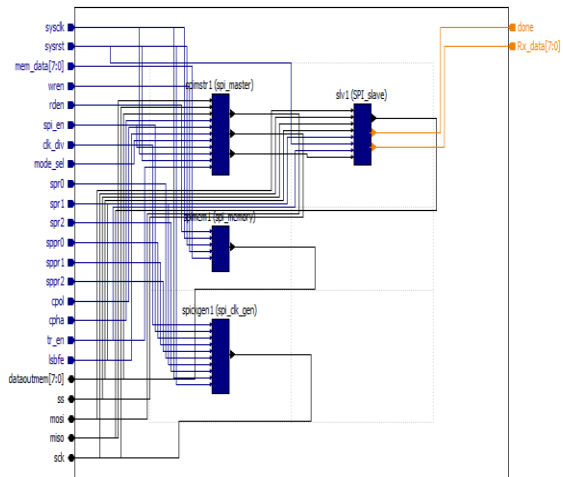


Figure 6. Hierarchal view of non-gated SPI.

Power, area (in terms of LUTs) and combinational delay of the top module is calculated taking system clock (sys_clk) = 100 MHz and SCK= 50 MHz. Fig.7 shows the results.

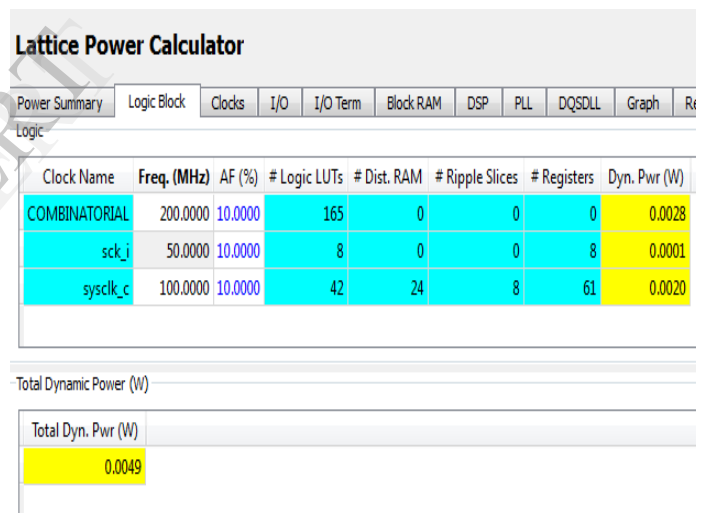


Figure 7 . Power and area calculation on FPGA.

As shown in Fig.7 total dynamic power contributed by the clocks ; combinational,SCK and sysclk is 4.9 mW and no of LUTs are 215. Maximum combinational path delay for this design is 6.917 ns. Minimum input arrival time before clock is 6.767ns and maximum output required time after clock: 5.513ns.

Fig.8 shows the hierachal view of Gated clock SPI module. This design includes on more block; Gated clock module (shown by black circle)

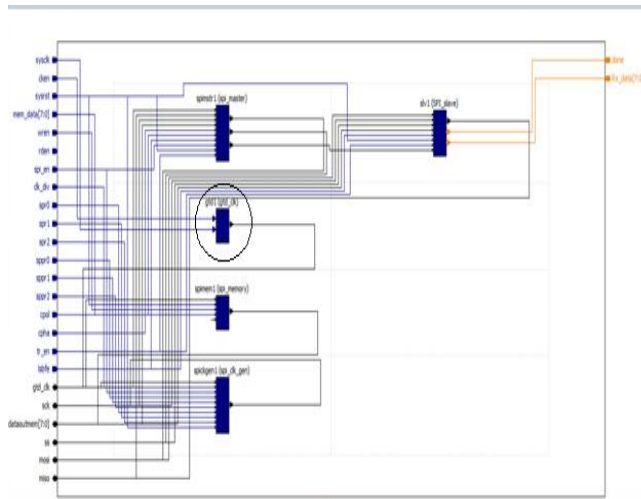


Figure 8. Hierachal view of Gated SPI.

Power, area (in terms of LUTs) and combinational delay of the top module is calculated taking gated clock (gtd_clk) = 100 MHz and SCK= 50 MHz. Fig.9 shows the results. Total dynamic power contributed by the clocks ; combinatorial,SCK and gtd_clk is 3.1 mW and no of LUTs are 165 . Maximun combinational path delay for this design is 6.840 ns. Minimum input arrival time before clock is 5.973ns and maximum output required time after clock: 5.513ns.

Lattice Power Calculator											
Power Summary	Logic Block	Clocks	I/O	I/O Term	Block RAM	DSP	PLL	DQSDDL	Graph	Report	
Logic											
Clock Name	Freq. (MHz)	AF (%)	# Logic LUTs	# Dist. RAM	# Ripple Slices	# Registers	Dyn. Pwr (W)				
COMBINATORIAL	200.0000	10.0000	135	0	0	0	0.0021				
sck_i	50.0000	10.0000	8	0	0	8	0.0001				
gtd_clk_c	100.0000	10.0000	22	0	0	36	0.0009				
Total Dynamic Power (W)								0.0031			

Figure 9 . Power and area calculation on FPGA.

Floorplan view of both the architectures is shown in Fig.10 and Fig.11.

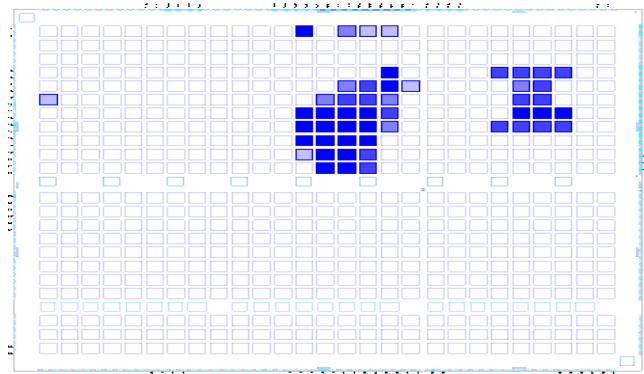


Figure 10. Floorplan view of Non-gated SPI Module.

In Fig. 10 and Fig.11, dark blue blocks are the PLC blocks and light blue blocks are the IO-logic blocks .

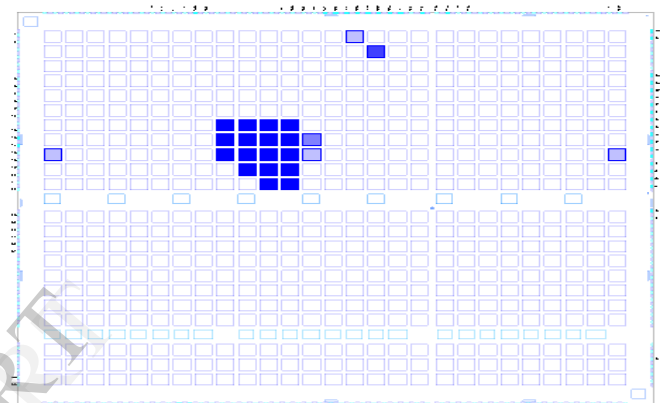


Figure 11. Floorplan view of Gated SPI Module

Table 1 shows the compariosn of Gated and Non-gated SPI module in terms of Dynamic power , area and combinational path delay on 2 FPGA devices.

Table 1. Compariosn between Gated and Non-gated modules.

FPGA Devices	Non-Gated SPI module			Gated SPI module		
	Dy.Power	Area	Delay	Dy.Power	Area	Delay
Lattice XP2 (For Power and Area) & Xilinx xa3s500ecpg132-4 (For Combinational path delay)	4.9mW	215 (LUTs)	6.917 ns	3.1mW	165 (LUTs)	6.840 ns

VI. CONCLUSION

The comparison between two modules; Gated and Non-gated SPI clearly shows that the overall dynamic power consumption of Gated clock SPI module is less as compared to the Non-gated SPI module also if area and combination path delay are to be considered choice will be Gated SPI module. A difference of 50 LUTs and a delay difference of 0.077 ns have been observed in the table. Hence, dynamic power as well as area and delay have been improved by RTL clock gating technique.

ACKNOWLEDGMENT

The authors wish thanks to all who were directly or indirectly provide their kind support.

REFERENCES

1. A.K Oudjida, M.L. Berandjia, R. Tiar, A.Liacha, K. Tahraoui, "FPGA implementation of I2C and SPI Protocols: A comparative study", IEEE, 2009.
2. Abhishek Godara, Suprita Chaudhary, "Power consumption reduction using RTL Clock Gating in AHB-Slave SPI-Master Architecture", International Journal Of VLSI and Signal Processing Applications, 2012.
3. M.K Md Arshad, U Hashim, Chew Ming Choo, "Characteristics of SPI Timing parameters of Optical Mouse Sensors", ICSE, 2006.
4. Frederic Leens, "An Introduction to I2C and SPI Protocols", IEEE, 2009.
5. Paul Myers, "Interfacing using Serial Protocols: SPI and I2C", EMRT Consultants, 2005.
6. SPI Block Guide V 03.06, Motorola Inc., 2003.
7. Frank Emmett, Mark Biegel, "Power Reduction through RTL Clock Gating", Automotive Integrated Electronics Corporation, 2000.
8. Xiaotao Chang, Mingming Zhang, Ge Zhang, Zhimin Zhang, Jun Wang, "Adaptive Clock Gating Technique For Low Power IP Core in SOC Design", IEEE, 2007.
9. SOC Central, "Design and Verification Techniques for Clock Gating", 2009.
10. Frederic Rivoallon, "Reducing Switching Power with Intelligent Clock Gating", Xilinx Paper, 2011.

IJERT