Name : Swaraj Singh

Email : swaraj.singh2021@vitstudent.ac.in

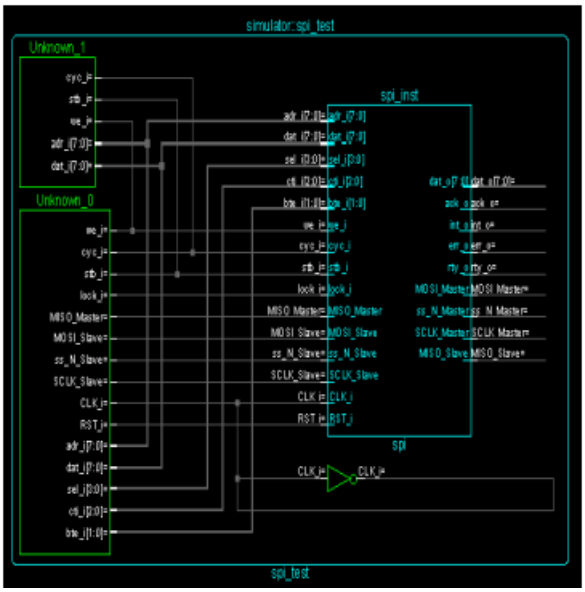
Final Report

I. INTRODUCTION

With the development of the IC manufacturing, the communication between hardware devices became particularly important. Every system includes some intelligent control, usually a Microcontroller Core. General-purpose circuits like LCD drivers, remote I/0 ports, RAM, EEPROM, or data converters. Application oriented circuits for communication interfaces and/or computation intensive task. So the communication between these modules are very important.

II. SPI Master

The Serial Peripheral Interface Bus or SPI bus is a synchronous serial data link standard, named by Motorola, that operates in full duplex mode. Devices communicate in master/slave mode where the master device initiates the data frame. During an SPI transfer, data is simultaneously transmitted and received. The serial clock line [SCK] synchronizes shifting and sampling of the information on the two serial data lines. The master places the information onto the MOSI line a half-cycle before the clock edge that the slave device uses to latch the data. Usually, the devices which based on SPI protocol are divided into master-device and slave-device for transmitting the data. The chip select signal and clock signal have to be generated by the master-device when the data exchange has been processed. As a multimaster/slave protocol, communications between the master and selected slave use the unidirectional MISO and MOSI lines, to achieve data rates over 1Mbps in full duplex mode. In addition to setting the clock frequency, the master must also configure the clock polarity and phase with respect to the data.



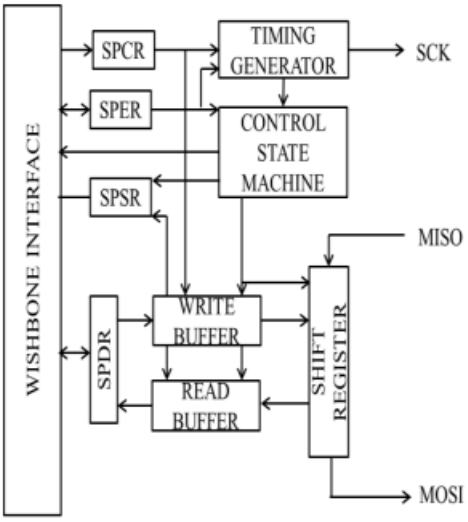


Fig. 1: Block Diagram

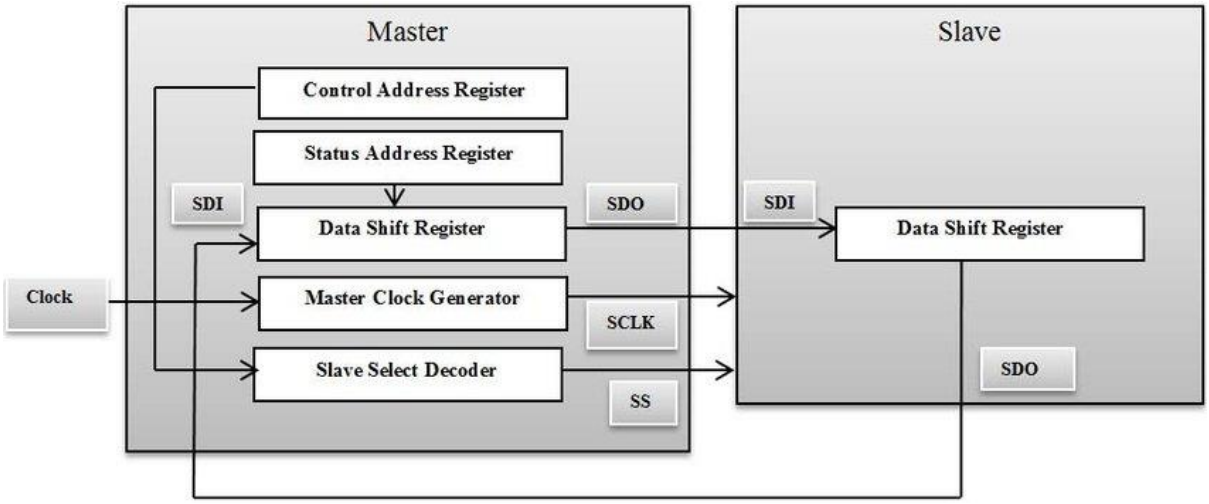
III. WISHBONE

The WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores is a flexible design methodology for use with semiconductor IP cores. Its purpose is to faster design reuse by alleviating System-on-Chip integration. Simple, compact, logical IP core hardware interfaces that require very few logic gates. The Wishbone Bus is an open source hardware computer bus intended to let the parts of an integrated circuit communicate with each other. The aim is to allow the connection of differing cores to each other inside of a chip. WISHBONE uses MASTER/SLAVE architecture. That means that functional modules with MASTER interfaces initiate data transactions to participating SLAVE interfaces. Here all internal WISHBONE logic is registered to the rising edge of the [clk_i] clock input. So, master clock decides the data to send or not. The active low asynchronous reset input [rst_i] forces the core to restart. All internal registers are preset and all statemachines are set to an initial state. The interrupt request output is asserted when the core needs service from the host system. When asserted, the cycle input [cyc_i] indicates that a valid bus cycle is in progress. The logical AND function of [cyc_i] and [stb_i] indicates a valid transfer cycle to/from the core. So when only the cycle input is high strobe signal is high. The strobe input [stb_i] is asserted when the core is being addressed. The core only responds to WISHBONE cycles when [stb_i] is asserted, except for the [rst_i], which always receive a response. The address array input [adr_i] is used to pass a binary coded address to the core. The most significant bit is at the higher number of the array. When asserted, the write enable input [we_i] indicates that the current bus cycle is a write cycle. When negated, it indicates that the current bus cycle is a read cycle. The data array input [dat_i] is used to pass binary data from the current WISHBONE Master to the core. All data transfers are 8 bit wide. The data array output [dat_o] is used to pass binary data from the core to the current WISHBONE Master. All data transfers are 8 bit wide. When asserted, the acknowledge output [ack_o] indicates the normal termination of a valid bus cycle.

SPI Master Top-Module:

Now, consider an SPI master top-module in a digital system. This module takes on the role of the master device in an SPI communication setup. Its functionality can be described in detail:

1. Clock Generation: The SPI master top-module is responsible for generating the serial clock (SCLK). This clock signal is crucial for synchronizing data transfer between the master and the connected slave devices.
2. Data Transfer Management: It manages the data transfer on both the MOSI and MISO lines. Data from the master is sent through the MOSI line, while data from the slaves is received through the MISO line.
3. Slave Selection: The SPI master controls the Slave Select (SS/CS) signal. By activating the SS/CS line for a specific slave, the master designates which device it wants to communicate with. This allows for communication with multiple slave devices on the same bus.



4. Communication Protocol Implementation: The SPI master top-module implements the SPI communication protocol, ensuring that data is transferred in the correct format and timing. This involves managing the start and stop of data transmission, as well as handling any protocol-specific configurations.
5. Error Handling: Depending on the design requirements, the SPI master module might include error detection and handling mechanisms to ensure reliable communication. This could involve checking for transmission errors and retransmitting data if necessary.
6. Integration with Higher-Level Logic: The SPI master top-module is likely to be integrated into a larger system, interfacing with higher-level logic or control modules. This integration ensures that SPI communication aligns with the overall functionality of the digital system.
7. Application Areas: SPI is commonly used in various applications, including interfacing with sensors, memory devices, display controllers, and communication between microcontrollers and peripheral devices. The SPI master top-module plays a crucial role in facilitating these diverse communication scenarios.

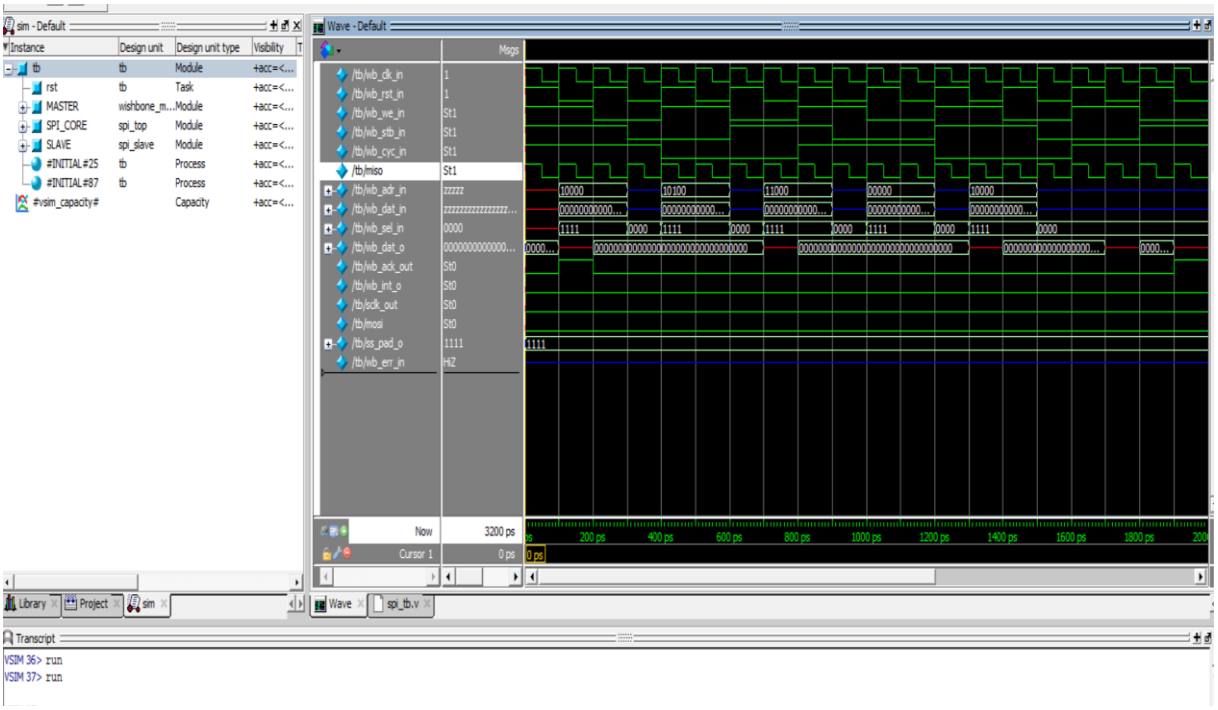
Conclusion: In summary, the SPI master top-module in a digital system is a fundamental component responsible for managing communication with peripheral devices using the SPI protocol. Its role encompasses clock generation, data transfer management, slave selection, protocol implementation, error handling, and integration with higher-level logic. Understanding the intricacies

of this module is essential for designing efficient and reliable embedded systems across a range of applications.

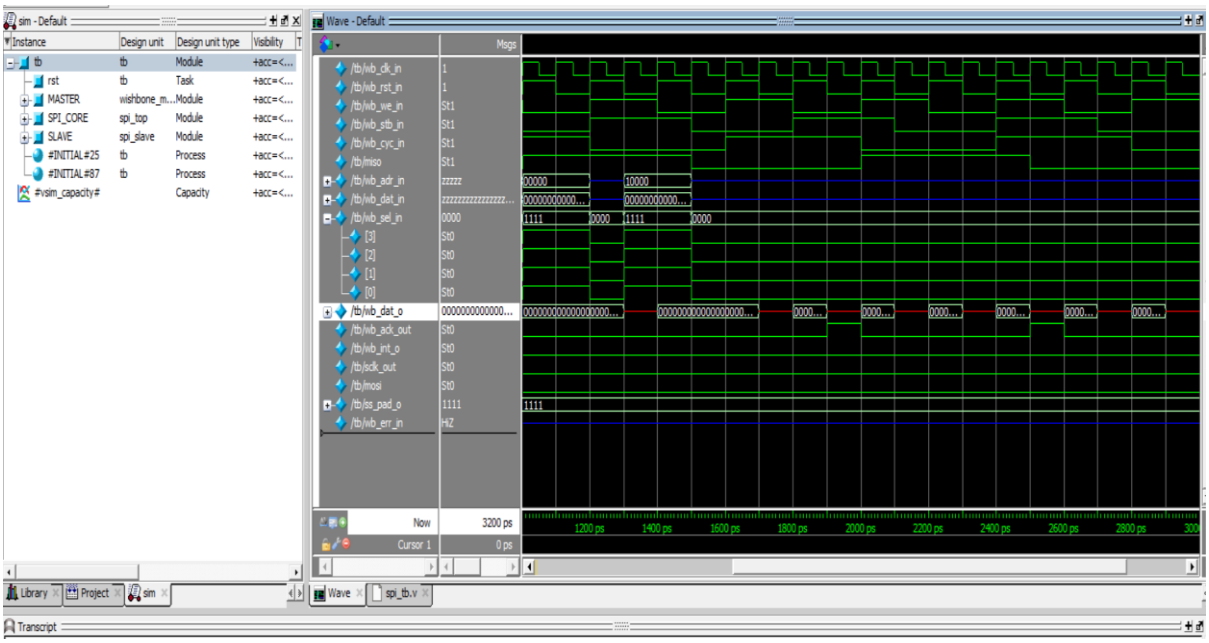
Output Waveform :

Top Module :

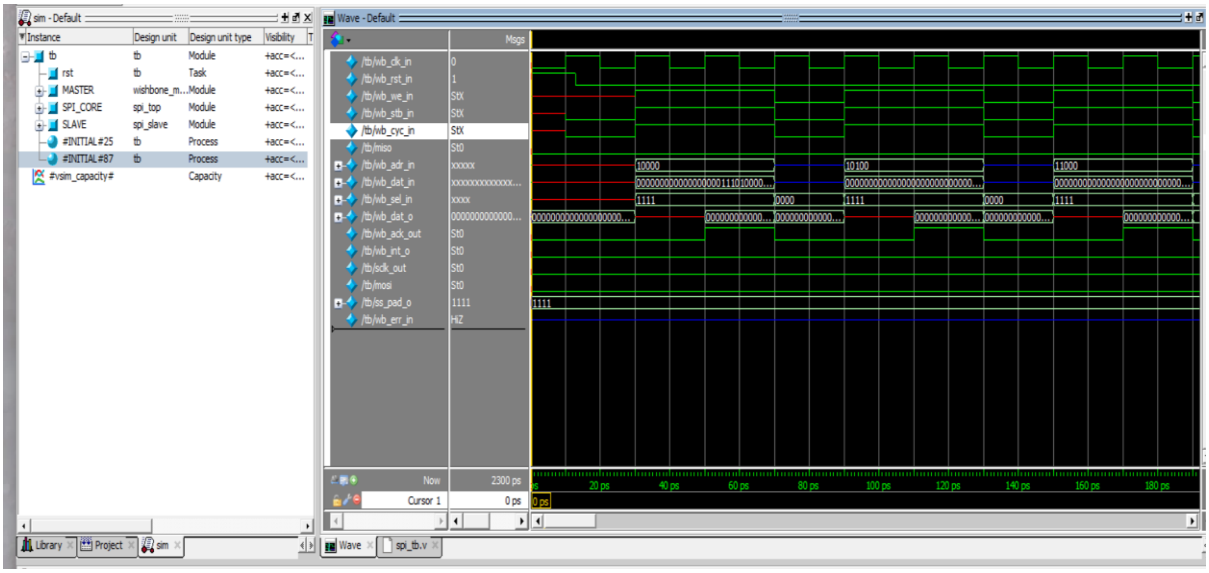
Scenario 1:



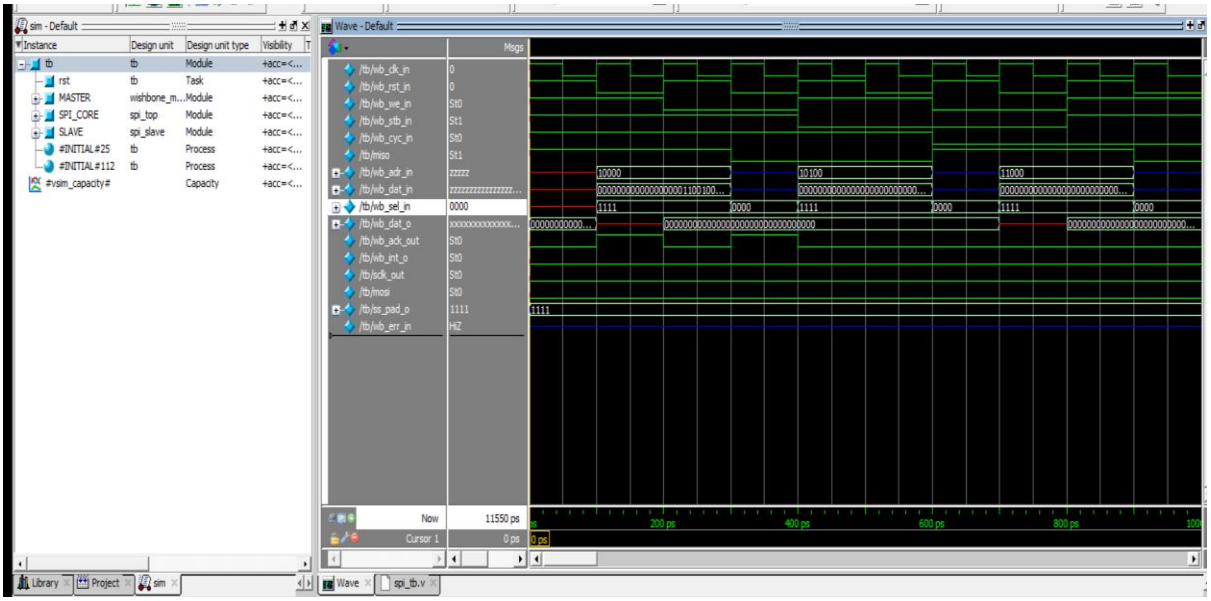
Scenario 2:



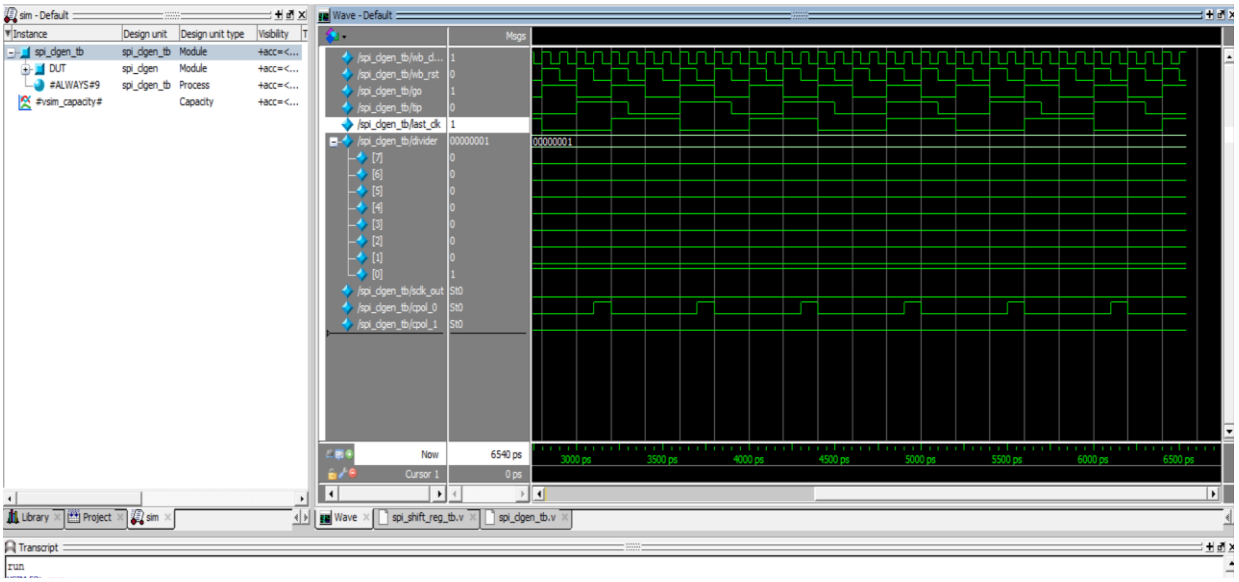
Scenario 3:



Scenario 4:



Clock Generator :



Shift Register: