**SPI INTERFACE Project**

Verilog Implementation and Design Flow

MAHMOUD GHAMRY

AUG.2024

Contents

1. Introduction …………………………………………………………………………………………………………… 1

2. Design Considerations ………………………………………………………………………………………….. 2

3. FSM ………………………………………………………………………………………………………………………. 3

4.CODING

4.1 RTL DESIGN………………………………………………………………………………………………… 4

4.2 TEST BENCH CODE ……………………………………………………………………………………… 10

4.2.1 Run Testbench in Questa sim 2021 …………………………………………………………. 12

5.USING VIVADO 2018

5.1 SEQUENTIAL ENCODING…………………………………………………… 14

5.1.1 Elaborated Design …………………………………………………………………………………. 14

5.1.2 Synthesis Design …………………………………………………………………………………….. 14

5.1.3 Implementation Design ………………………………………………………………………….. 15

5.1.4 REPORTS…………………………………………………………………………………………………. 16

5.2 GRAY ENCODING……………………………………………………………. 17

5.2.1 Elaborated Design …………………………………………………………………………………. 17

5.2.2 Synthesis Design …………………………………………………………………………………….. 17

5.2.3 Implementation Design ………………………………………………………………………….. 18

5.2.4 REPORTS…………………………………………………………………………………………………. 19

6 BITSTREAM ………………………………………………………………………………………………………………….. 20

7. ADDTION FILES

7.1 DO FILE…………………………………………………………………………………………………………. 20

7.2 Constraints…………………………………………………………………………………………………… 20

1. **Introduction**

The Serial Peripheral Interface (SPI) is a widely used synchronous serial communication protocol that enables the transfer of data between a microcontroller and peripheral devices such as sensors, memory chips, and display modules. Characterized by its simplicity and speed, SPI operates using a master-slave architecture where a master device controls multiple slave devices through four primary signals: MISO (Master In Slave Out), MOSI (Master Out Slave In), SCLK (Serial Clock), and SS\_n (Slave Select).

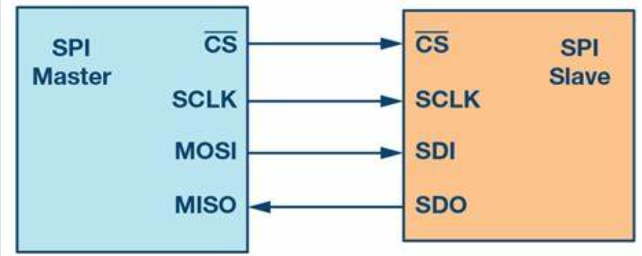
In the realm of digital design, implementing SPI communication using Verilog—a hardware description language (HDL)—allows designers to create flexible and efficient custom interfaces tailored to specific requirements. Verilog, with its powerful constructs for describing and simulating digital systems, enables precise control over the timing and functionality of the SPI protocol.

The essence of SPI communication in Verilog involves the creation of modules that encapsulate the behavior of both master and slave devices. These modules manage the data flow, handle synchronization issues, and ensure that data integrity is maintained throughout the communication process. The implementation encompasses the finite state machines (FSMs) for state transitions, shift registers for serial-to-parallel and parallel-to-serial data conversion, and control logic for signal synchronization.

This document explores the fundamental principles of the SPI protocol and provides a detailed guide to implementing both SPI master and slave modules using Verilog. Through practical examples and code snippets, we illustrate the key concepts and techniques required to achieve reliable SPI communication, making it an invaluable resource for anyone looking to integrate SPI interfaces into their digital designs.

1-20

**2.** **Design Considerations**



*Figure 2-1:* **SPI PROTOCOL**

dout

*Figure 2-3:* **SPI RAM**

RAM

Tx\_vaild

Rx\_vaild

din

rst\_n

clk

*Figure 2-2:* **SPI SLAVE**

Tx\_data

Tx\_vaild

Rx\_vaild

Rx\_data

rst\_n

clk

SS\_n

MISO

MOSI

SPI Slave

صورة تحتوي على نص, رسم بياني, لقطة شاشة, خط

تم إنشاء الوصف تلقائياً

2 -20 *Figure 2-4:* **SPI WRAPPER**

3.FSM

SS\_n =1

IDLE

READ

DATA

SS\_n =1

SS\_n =1

SS\_n =1

SS\_n =0

WRITE

SS\_n =0

$ mosi =1

READ

ADD

SS\_n =0

$ mosi = 0

CHK

CMD

SS\_n =0

$ mosi =1

3-20

4 Coding

4.1 RTL DESIGN

صورة تحتوي على نص, لقطة شاشة, قائمة طعام, الخط

تم إنشاء الوصف تلقائياً

*Figure 4-1:* **SPI SLAVE**

4-20

صورة تحتوي على نص, لقطة شاشة

تم إنشاء الوصف تلقائياً

*Figure 4-2:* **SPI SLAVE**

5-20

صورة تحتوي على نص, لقطة شاشة, قائمة طعام

تم إنشاء الوصف تلقائياً

*Figure 4-3:* **SPI SLAVE**

6-20

صورة تحتوي على نص, لقطة شاشة, قائمة طعام

تم إنشاء الوصف تلقائياً

*Figure 4-4:* **SPI SLAVE**

7-20

صورة تحتوي على نص, لقطة شاشة, برمجيات, الخط

تم إنشاء الوصف تلقائياً

*Figure 4-5:* **SPI SLAVE**

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

*Figure 4-6:* **SPI RAM**

8-20

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

*Figure 4-7:* **SPI RAM**

صورة تحتوي على نص, لقطة شاشة, برمجيات, برامج الوسائط المتعددة

تم إنشاء الوصف تلقائياً

*Figure 4-8:* **SPI WRAPPER**

9-20

4.2 Test bench

صورة تحتوي على نص, الخط, لقطة شاشة

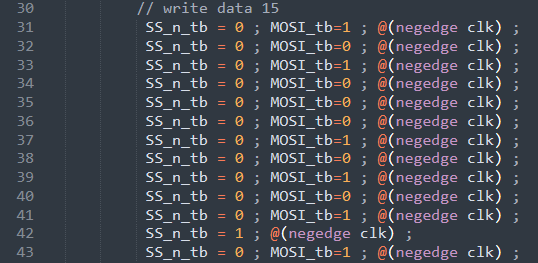
تم إنشاء الوصف تلقائياً

*Figure 4-2-1:* **SPI Test bench**

**صورة تحتوي على نص, لقطة شاشة, قائمة طعام, الخط

تم إنشاء الوصف تلقائياً**

*Figure 4-2-2:* **SPI WRITE ADDERSS**

****

*Figure 4-2-3:* **SPI WRITE DATA**

10-20

صورة تحتوي على نص, لقطة شاشة, قائمة طعام, الخط

تم إنشاء الوصف تلقائياً

*Figure 4-2-4:* **SPI READ ADDRESS**

**صورة تحتوي على نص, لقطة شاشة, قائمة طعام, الخط

تم إنشاء الوصف تلقائياً**

*Figure 4-2-5:* **SPI SLAVE**

11-20

4.2.1 Run Testbench in Questa sim 2021

صورة تحتوي على نص, لقطة شاشة, برمجيات, عرض

تم إنشاء الوصف تلقائياً

*Figure 4-2-1:* **WIRTE ADDRESS IN 0xd8**

**صورة تحتوي على لقطة شاشة, الإلكترونيات, دائرة كهربائية

تم إنشاء الوصف تلقائياً**

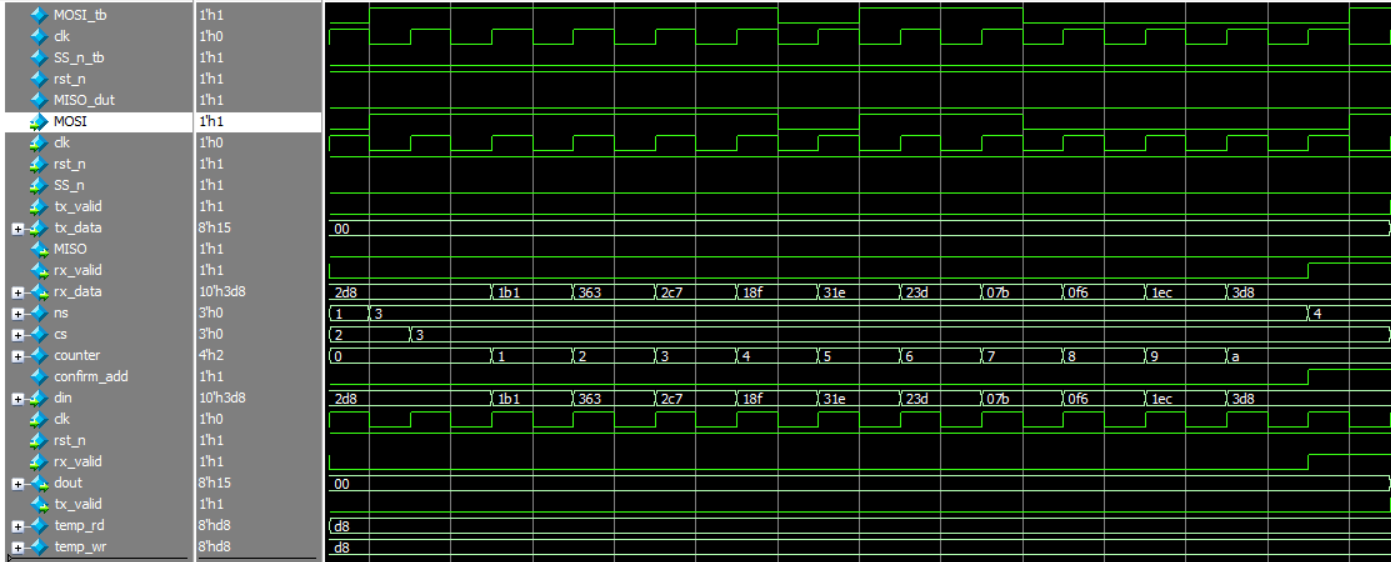
*Figure 4-2-2:* **WIRTE DATA 0x15**

12-20

**صورة تحتوي على لقطة شاشة, دائرة كهربائية

تم إنشاء الوصف تلقائياً**

*Figure 4-2-3:* **READ ADDRESS FROM 0xd8**



*Figure 4-2-4:* **READ DATA FROM MEMEORY**

صورة تحتوي على لقطة شاشة, خط, التلون

تم إنشاء الوصف تلقائياً

*Figure 4-2-5:* **MISO OUTPUT**

13-20

5.USING VIVADO 2018

5.1 SEQUENTIAL ENCODING

5.1.1 Elaborated Design

صورة تحتوي على نص, رسم بياني, خطة, خط

تم إنشاء الوصف تلقائياً

*Figure 5-1-1:* Elaborated Design

5.1.2. Synthesis Design

صورة تحتوي على نص, رسم بياني, خطة, خط

تم إنشاء الوصف تلقائياً

*Figure 5-1-2:* Synthesis Design

14-20

5.1.3. Implementation Design

صورة تحتوي على رسم بياني, نص, خطة, رسم تقني

تم إنشاء الوصف تلقائياً

*Figure 5-1-3-1:* Implementation Design

صورة تحتوي على نص, رسم بياني, مستطيل, خط

تم إنشاء الوصف تلقائياً

*Figure 5-1-3-2:* Implementation Design

15-20

5.1.4.REPORT

صورة تحتوي على نص, الخط, خط, لقطة شاشة

تم إنشاء الوصف تلقائياً

*Figure 5.1.4.1:* TIME

**صورة تحتوي على نص, لقطة شاشة, خط, الخط

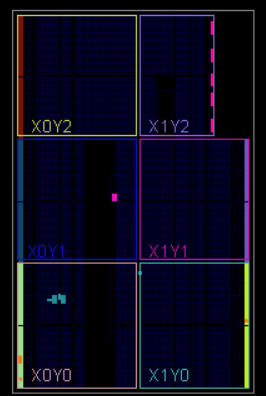
تم إنشاء الوصف تلقائياً**

*Figure 5.1.4.2:* UTILIZATION

**صورة تحتوي على نص, إيصال, لقطة شاشة, خط

تم إنشاء الوصف تلقائياً**

*Figure 5.1.4.3:* SEQUENTIAL ENCODING



16-20 *Figure 5.1.5* DEVICE

5.2 GRAY ENCODING

5.2.1 Elaborated Design

صورة تحتوي على رسم بياني, نص, خطة, رسم تقني

تم إنشاء الوصف تلقائياً

*Figure 5-2-1:* Elaborated Design

5.2.2 Synthesis Design

صورة تحتوي على نص, رسم بياني, خطة, خط

تم إنشاء الوصف تلقائياً

***Figure 5-2-2:* Synthesis Design**

17-20

5.1.3. Implementation Design

**صورة تحتوي على نص, رسم بياني, خطة, رسم تقني

تم إنشاء الوصف تلقائياً**

*Figure 5-2-3-1:* Implementation Design

**صورة تحتوي على مستطيل, رسم بياني, خط, خطة

تم إنشاء الوصف تلقائياً**

*Figure 5-2-3-2:* Implementation Design

18-20

**5.2.4 REPORT**

**صورة تحتوي على نص, الخط, خط, لقطة شاشة

تم إنشاء الوصف تلقائياً**

**Figure 5-2-4-1: TIME**

**صورة تحتوي على نص, خط, رقم, الخط

تم إنشاء الوصف تلقائياً**

*Figure 5.2.4.2:* UTILIZATION

صورة تحتوي على نص, لقطة شاشة, خط, إيصال

تم إنشاء الوصف تلقائياً

*Figure 5.3.4.3:* GRAY ENCODING

صورة تحتوي على لقطة شاشة, نص, ميدان/ مربع

تم إنشاء الوصف تلقائياً

19-20 *Figure 5.2.5* DEVICE

6 BITSTREAM CHECK

صورة تحتوي على نص, لقطة شاشة, الخط, خط

تم إنشاء الوصف تلقائياً

*Figure 6* CHECK

**7.ADDTION FILES**

7.1 DO FILE

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

*Figure 7.1 DO FILE*

7.2 Constraints

صورة تحتوي على نص, لقطة شاشة, الخط

تم إنشاء الوصف تلقائياً

*Figure 7.2 CONSTRAINTS FILE*

20-20