IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016.

Transfer of NOR Flash Instructions through Serial Peripheral Interface Bus

Laxmi[1], Dr. Yogish H K[2]

M.Tech Student, Department of ISE (CNE), SJBIT, Bengaluru, India[1]

Professor, Department of ISE (CNE), SJBIT, Bengaluru, India[2]

Abstract: This paper focuses on the analysis of NOR flash and Serial Peripheral Interface (SPI) bus. In this paper we are dealing with the testing feature of the NOR flash memory and developing the architecture of validation suite. The communication is achieved via SPI bus between host and target. Host and Target are microcontroller and FPGA. In between these two SPI bus acts as an interface. NOR flash is a kind of permanent technology that stores the data or information even though the power is off. There are two types of Flash memory, such as NOR flash memory and NAND flash memory. NOR flash is used because it has execute in place feature and read operation is faster than NAND flash memory. Also comparison in done in between three approaches like Extended SPI approach, Quad approach and Octal approach.

Keywords: NOR Flash, SPI bus, Host, Target, execute in place, NAND Flash, communication.

I. INTRODUCTION

The NOR Flash memory has a feature of non volatile technology that is used to erase the information in blocks [1]. The information contained on the flash memory should be removed first before the data can be programmed or stored on the chip. NOR Flash retains the information for a very long period of time even though the power is not there [3]. The existence of Flash memory is from Erasable Programmable Read Only Memory (EPROM) and Electrically Erasable Programmable Read Only Memory (EEPROM). Flash memory removes the data in block level and data can be reprogrammed at the byte level.

NOR memory uses lower standby control and is very easy for implementation of the code. NOR flash is speed in terms of reading an array of bytes from random address when compared to NAND flash. On consideration with the architecture of the NOR and NAND flash, the cells in NOR flash are arranged in parallel to the bit lines. This way of configuration allows the cells to program and read individually. In NAND flash, the way of ordering is in series that is similar to the NAND gate. FPGA is an Integrated circuit (IC) which is designed by the customer after manufacturing [9], hence the name field programmable. FPGA hardware includes programmable logic blocks. Logic blocks are the memory components. For ex: Flip-Flops or memory blocks [2].

Serial Peripheral Interface (SPI) is the synchronous serialized communication that is made use for very short distance communication primarily in embedded applications. SPI interface was developed by Motorola Company. SPI drivers are proficient to communicate in full duplex mode that uses Master – Slave architecture [12] or Host and Target architecture. Out of I2C and SPI protocol SPI bus is efficient to transmit the data very speed [11]. SPI bus uses one master communication protocol. There is one essential device that initiates the message with the slaves.

II. LITERATURE SURVEY

This section explains about the previous studies that have been approved in the field of NOR flash and SPI bus. Memory can be divided into secondary memory and primary memory. The paper highlights the NOR flash memory [15]. The memory can be also be divided as Volatile memory and Non-Volatile memory as shown in the Fig. 1. Non-volatile memory keeps the information even if the power is lost. With the help of ROM , we can read the information but cannot write [6]. The memory is stored fixed during the manufacturing. Programmable Read Only Memory (PROM) is the memory that can be modified by the user only once. The user buys empty PROM and enters the data using a PROM program. Erasable Programmable Read Only Memory (EPROM) is used to clear the data and re-use it. Electrically Erasable Programmable Read-Only Memory (EEPROM) is used to code and remove electrically [5]. It can be erased and reprogrammed as many times as per the user requirement. These two memories are unique in functionality. NOR is fast and costly whereas NAND stores high capacity data. An example of NOR flash memory is cell phone and examples of NAND is MP3 players, USB cables etc. Volatile memory includes RAM, Static RAM [7] and Dynamic RAM. SPI is a common technology that is used for communication with peripheral devices so that we can transmit data fast considering the real time constraints. RAM is called as read or write memory. With the assist of RAM, the programmer can read or write the information. Flip flop is an example of Static RAM (SRAM) which makes use of transistors. Dynamic RAM requires

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016.

refreshing of memory cells. SRAM is faster when compared to DRAM since, SRAM uses low access time and DRAM has an advantage of low cost.

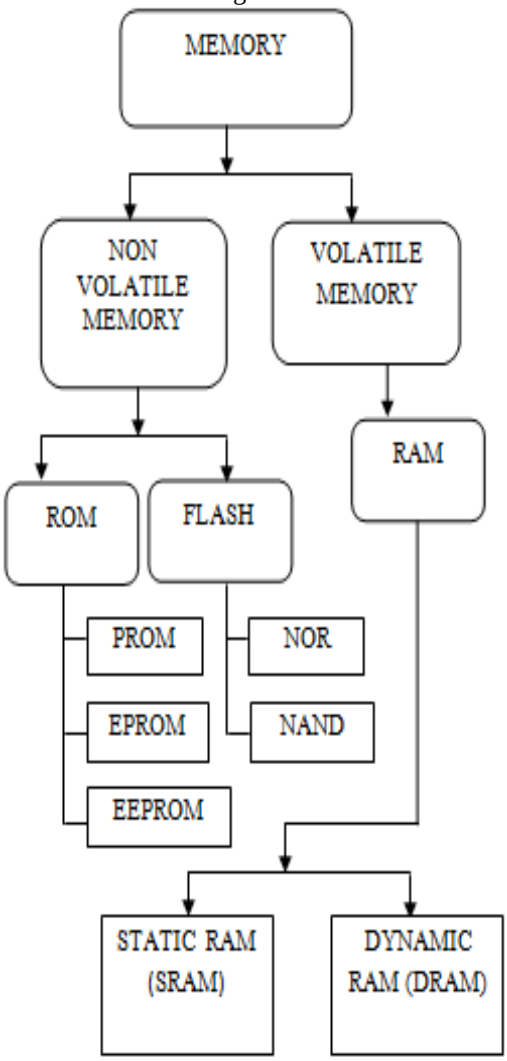

Fig1. Memory Classification Diagram

SPI technology is built to vary the parallel interfaces so that we need not direct parallel bus around PCB and also it includes the feature of providing high speed data transmission between the devices. The SPI bus is a simple yet sophisticated four wire serial communication protocol which is used by many of the microcontroller or microprocessor peripheral devices that enables the controllers and peripheral systems to have a communication with each other. In SPI, the data is shifted in or out one at a time and transfer the data from master device to or from one or more slave devices with high speed.

III. IMPLEMENTATION

In this paper we are achieving the data transmission between the host(microcontroller) and the target(FPGA). The communication is mainly through the SPI as shown in Fig. 2. The microcontroller acts as a host and FPGA acts as a target. We cannot connect the microcontroller and FPGA directly because they need additional connectors to connect both these two devices. Three modes of data transfer is possible: Extended SPI, Quad and Octal.

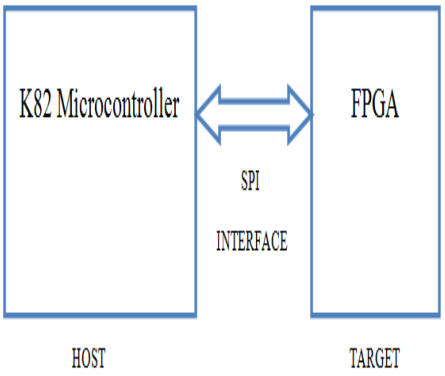

Fig.2. Block Diagram of System Model

A. Extended SPI approach

A group of commands are sent from host to target commonly known as a SPI master. SPI host interacts with the slave target through four signal lines they are chip select, serial clock, data select, serial input and serial out.

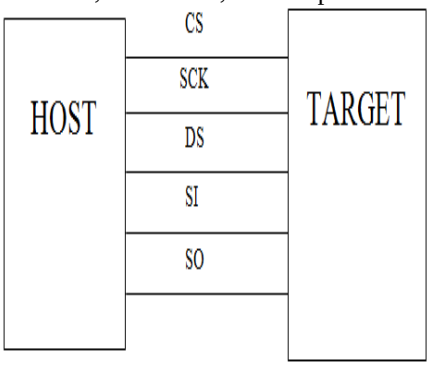

Fig. 3. Extended SPI approach

B. Quad SPI approach

Performance is increased with the help of this approach. In this approach four data lines are used for transfer of data and receiving the data. Single transfer mode and dual transfer mode are supported in this approach. The control byte is clocked in one or two clock cycles.

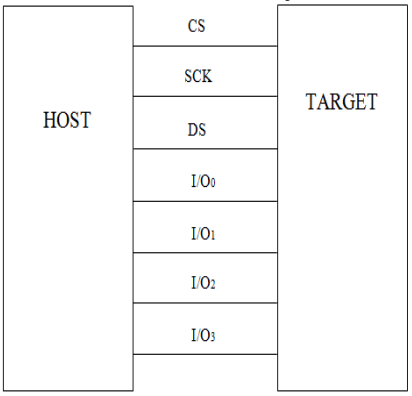

Fig. 4. QuadSPI approach

C. OctalSPI approach

This approach is for increasing the efficient transfer rate of data between microcontroller and FPGA. Host interacts with the slave target through 8 input lines and output lines.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016.

As shown in the Fig. 5, CS is the chip select, SCK is the serial clock, DS is the data strobe and I/O is input/outputs.

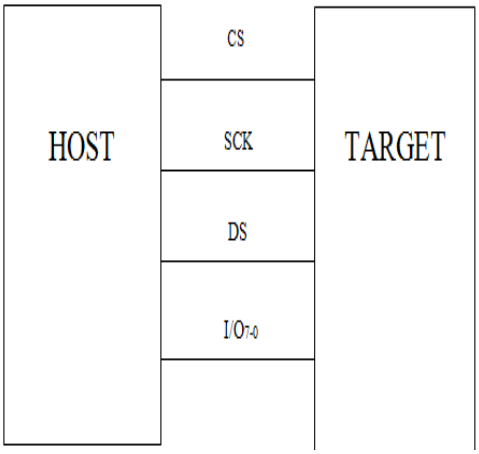

Fig. 5. Octal SPI approach

D. Testing an application

It is the testing approach of the paper with which we can get the outputs by validating or testing the scenarios such as " test all the instructions", "test individual test cases" or "displaying all the test cases".

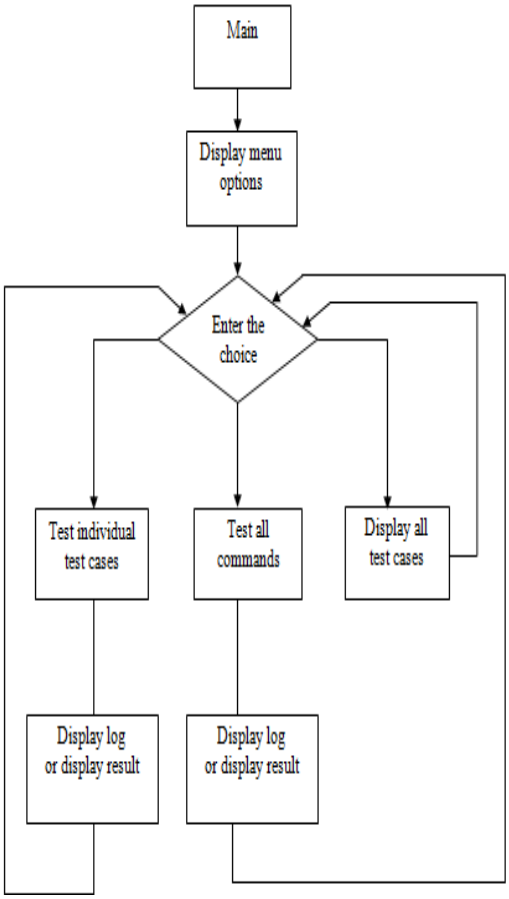

Fig. 6. Flow chart of testing approach

E. Description of few instructions

- Read Instructions are used to read the data from the device on activation of read command. To execute the read operation, the chip select pin should be activated and suitable opcodes are used. When subsequent opcode is clocked in, then send three address bytes. Later send the data bytes to the specified address.
- Erase Instructions are used to clear the data using the instruction erase command. There is a feature of erasing a number of blocks or we can erase a complete block of data. To use erase command, first we need to enable the write command. If write command is not activated, erase command will not be activated.
- Sector Protect Instructions are for protecting the bit sectors and also it helps in maintaining the protection of the device. When the device is reset, the sector protection instruction resets the level to logical "1". This instructions are primarily used for confirming that all the sectors are protected.

F. Method Prototypes used in implementation

The method prototype for Read Array, Block Erase and Chip Erase are shown in Fig. 7, for Extended SPI, Quad and Octal approaches.

```
void ReadArray_SPI (unsigned int ui32StartAddr,
unsigned int len, unsigned char *ui8buff)
{
    ----------
    ----------
}
void BlockErase_QPI (unsigned int ui32StartAddr)
{
    /* Permit Write Enable logic before program */
    ----------
    ----------
 }
void ChipErase_OPI(unsigned int ui32StartAddr)
{
    /* Permit Write Enable logic before program */
    ----------
    ----------
 }
```

Fig. 7. Screenshot of method prototype

IV. RESULT ANALYSIS

The Result Analysis deals with the experimental outcomes of the paper using Vivado Tool. Vivado 2015.4 is used for analyzing the performance of Extended SPI approach, Quad approach and Octal approach. Vivado2015.4 is a simulation tool that virtualizes the outputs through timing diagrams. Every instructions used in implementation have standard and fixed timing diagrams as per the client requirements.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016.

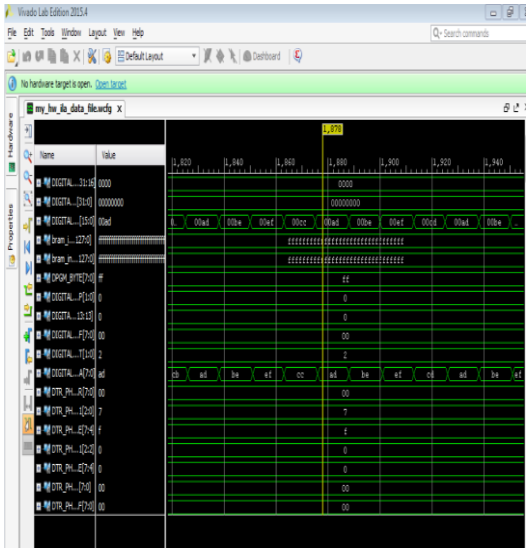

Fig.8.Simulation results of Extended SPI approach

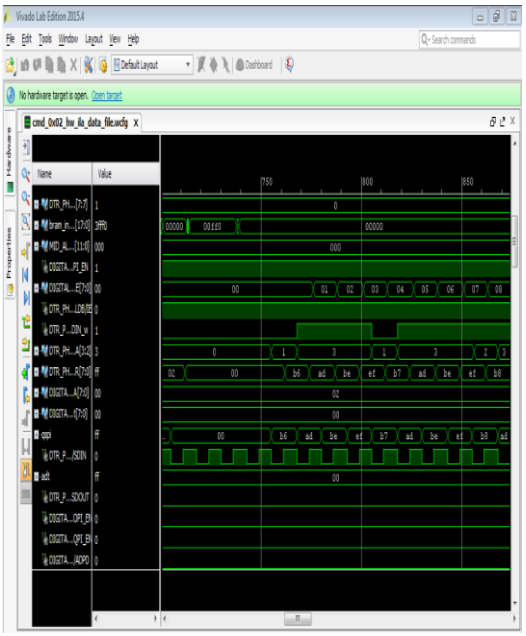

Fig. 9. Simulation results of Quad approach

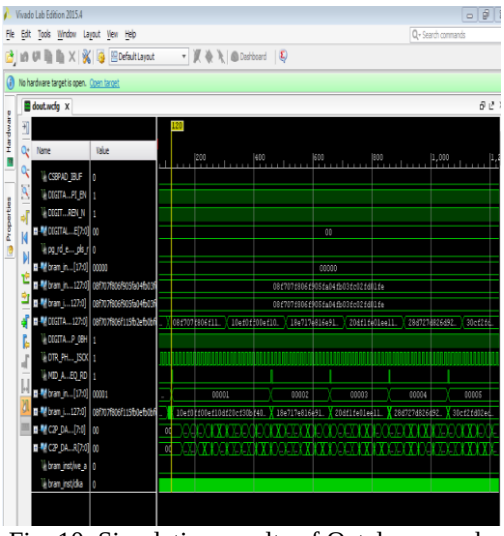

Fig. 10. Simulation results of Octal approach

Extended SPI approach is slower than other two approaches such as, Quad approach and Octal approach. Quad approach is faster than Extended SPI but it is slower than Octal approach. All the instructions used in implementation have fixed timing diagrams. If the timing diagrams are same in both the requirements and in Vivado tool then it prints success else failure.

V. CONCLUSION

In this paper, we have studied about communication of SPI bus and testing the feature of NOR are flash instructions and three approaches compared with the help of simulation tool, Vivado. The throughput of the Octal approach is very fast when compared to other two approaches. Simulation result of Extended SPI approach is shown in Fig. 8, similarly Fig. 9. and Fig. 10 displays the output of Quad and Octal approach. Extended SPI approach is slower because the input and output lines are less in number. Some of the instructions used here are Read instruction, Program and Erase instruction, Security instruction and Status or Control Register instruction, Protection instruction which increases the performance of SPI bus.

ACKNOWLEDGMENT

I express my sincere gratitude to **Mr. Sangameshwar Pujari,** Manager of Power Square India Pvt. Ltd. for guiding me throughout my project and giving me such a wonderful experience as an intern.

REFERENCES

[1]. P. Pavan, R. Bez, P. Olivo, and E. Zanoni, “Flash memory cells—An overview,” Proc. IEEE, vol. 85, pp. 1248–1271, Aug. 1997.
[2]. G. Casagrande, “Flash memory testing,” in Flash Memories, P. Cappelletti et al., Ed. Norwell, MA: Kluwer, 1999.
[3]. S P. Cappelletti and A. Modelli, “Flash memory reliability,” in Flash Memories, P. Cappelletti et al., Ed. Norwell, MA: Kluwer, 1999.
[4]. A. Modelli, A. Manstretta, and G. Torelli, “Basic feasibility constraints for multilevel CHE-programmed Flash memories,” IEEE Trans. Electron Devices, vol. 48, pp. 2032–2042, Sept. 2001.
[5]. J.M.Portal & al., "Floating-Gate EEPROM Cell Model Based on MOS Model 9", IEEE Int'l Symp. on Circuits and Systems, 2002.
[6]. F. Leens, "Solutions for SPI Protocol Testing and Debugging in Embedded System," Byte Paradigm's White Paper, pp. 1-9, Revision 1.00, June 2008.
[7]. A. K. Oudjida, M. L. Berrandjia, R. Tiar, A. Liacha, K. Tahraoui, "FPGA implementation of I2C & SPI protocols: A comparative study," in Proc. 16th IEEE International Conference on Electronics, Circuits, and Systems, pp.507-510, Dec. 2009.
[8]. A. K. Oudjida, M. L. Berrandjia, A. Liacha, et al. "Design and test of general-purpose SPI master/slave IPs on OPB bus," 2010 7th International Multi-Conference on Systems Signals and Devices (SSD). 2010, pp. 1-6.
[9]. A.N. Gaidhane and M.P. Khorgade, "FPGA Implementation of Serial Peripheral Interface of FlexRay Controller," in Proc. 13th International Conference on Computer Modelling and Simulation (UKSim), pp.128-132, Apr. 2011.
[10]. N.Q. B. M. Noor and A. Saparon, "FPGA implementation of high speed serial peripheral interface for motion controller," in Proc. 2012 IEEE Symposium on Industrial Electronics and Applications (ISIEA), pp.78-83, Sept. 2012.
[11]. Jim Elliott and Eunseung Jung, "Ushering in the 3D Memory Era with V-NAND," in Flash Memory Summit 2013.