International Journal of Pure and Applied Mathematics
Volume 118 No. 24 2018
ISSN: 1314-3395 (on-line version)
url: http://www.acadpubl.eu/hub/
Special Issue

http://www.acadpubl.eu/hub/

Functional Verification of Inter Integrated Circuit (I2C) Using Advanced Verification Methodology

G.Renuka[1] , V.Ushashree[2] , P .Chandrasekhar Reddy[3]
[1] Department of Electronics & Communication Engg.,
SREC , Warangal, Telangana, India
[2] Department of Electronics & Communication Engg.,
JBIET, Hyderabad, Telangana, India.
[3] Department of Electronics & Communication Engg.,
JNTU, Hyderabad, Telangana, India.
renu_manaswith@yahoo.com

April 29, 2018

Abstract

The technology progressions leads to place large number of circuits on a silicon chip which forms verification a challenging job than previously. A very great number of resources are required since almost 70% of the overall design phase is meant for verification. Advanced Verification Methodology provides reusable verification environment and established to give a well-structured verification that does not affect the DUT. This Research paper gives the reusability of Inter Integrated circuit (I2C) using AVM and shows how test cases are executed and the verification environment is created.

***Key Words*:**I2C ,Environment, SoC, Testbench SystemVerilog, Verification, AVM.

1 Introduction:

A bus is a link between the modules of a structure , e.g .- a link among CPU and main memory, between the peripheral devices, I/O ports and memory. Different varieties of buses can used for data transferal. They are of I2C, WISHBONE, PCI, SPI, AMBA and USB. Each and every bus will have distinctive bus speed. I2C is Inter Integrated circuit and was developed in 1980s by PHILIPS SEMICONDUCTOR for use in television.I2C is a modest, 2-wire interconnect consist of a serialized data channel and a clock. It supports device addressing that could shorten the system design and then permits additional modules to be added effortlessly - up to maximum of 112 devices with a limitations of the regular address space of 7-bit[1]. The I2C bus is easy to implement and is also cost effective. Simplicity and flexibility are the key features that make I2C attractive to various applications . The Coverage driven verification and object oriented feature of AVM are efficient for functional verification. Verification is a procedure acclimated for ensuring that the backdrop of the architecture assemblage are conserved above-mentioned to the mapping into silicon. And this process is active in alongside to the architecture procedure.So at every stage of the design it is possible to can check out the bugs.The key purpose of anatomic analysis is award failures ,detecting bugs and acclimation before they are given into the IC. As electronics bazaar is alteration rapidly and its advance actuality all-inclusive it induces engineers to go for circuitous IC architecture and stuffing them into baby spaces. So systems on chip (SOC) are being developed and 70 % of architecture accomplishment leads to verification.

Advanced Verification Methodology (AVM) is a functional verification methodology for design entities. It is created on UVM .Its Class Library affords the building modules, test environments and verification components. It takes system Verilog as its language and all the main simulation vendor supports it. AVM may be an open wellspring technique to utilizing System-Verilog. It will be intended basically for confirmation about scholarly properties Furthermore test bench segments with the goal that test benches need aid reusable What's more confirmation code is more convenient. Each confirmation part takes after a steady building design for simulating, checking Furthermore gathering utilitarian scope

report card. Those confirmation nature's domain formed through System-Verilog might make fluctuate depending upon implementer.

2 I2C PROTOCOL

I2C gives chip-to-chip serial exchanges utilizing best two lines to an interface as in fig 1. The two lines in the I2C bus convey one bit of address selection, direction control, and data items at a time[2]. The data (SDA) line transmits the data items , while (SCL) the Clock line coordinates the transmitter and receiver during the transmission. Device that uses I2C communication protocol needs very limited pins to implement the similar function as their huge parallel interface counterparts. The I2C bus has 3 modes of operation: standard mode (0 to 100 kbps), fast mode (0 to 400 kbps) and high-speed mode (0 to 3.4 Mbps)

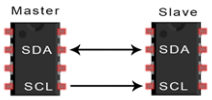

Fig 1:I2c signals

Properties of I2c bus controller

1) The speed bit is eventually determined by Baud rate of 100kbps or 400kbps .

2) Slave lets SCL to be low to suspend master

3) module is busy or not is determined by Busy bit.

4) Single master controller supports only master mode But not arbitration .

5 1 byte data transmission or reception and 3 bit addressing mode are supported.

I2C Specifications

Those I2C transport physically comprises from claiming two bidirectional dynamic wires- serial information line, serial clock offering What's more An ground association. as in figure1. Each and

every device coupled to the bus has its specific distinctive address and will performance as a receiver or as a sender, depending on the necessary functionality. The I2C protocol is advised as a multi-level master bus. This bus Master is an IC which initiates a abstracts alteration on the bus protocol and all added ICs are advised as Slaves. Master can affair an 'Attention' arresting to every one of the accompanying accessories accepted as a START[3]. Then the Master I2C will send the ADDRESS of the bondservant accessory that needs to be accessed. The Write or Read operational signal data or bit can also be sent with the ADDRESS bits. All the modules or devices associated on the bus will compare their own address with the sent address bits and if it doesn't match, they just wait untill the bus is free and if the address are equals; chip will produce a response known to be as Acknowledgement (ACK) signal[4]. By getting an acknowledgement signal , master I2C starts transmitting the DATA items. Each data byte may be of 8 bits long. An acknowledge signal monitors each transmitted byte. After transmission has been done, the Master I2C will drive the STOP bit as in figure 2.

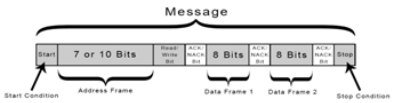

Fig 2 : I2C Communication Frame Format

I2C Communication Procedure

The communication procedure is as follows:

Step 1: It Checks if there is any bus movement occurrence or not. If both of SCL and SDA lines are high then the bus can be free. If theie is availablility of bus then master creates START condition.

Step 2: SCL offers clock bits to all of the ICs linked through the bus as a reference clock bit or signal. The data items on the data wire (SDA) line must be valid at the time when the clock signal (SCL) changes from 'low' to 'high' voltage [5].

Step 3: Address of each and every module is placed serially on the SDA line.

Step 4: One bit is placed on to the SDA wire line to know whether the data item is to be transferred or to be received from the slave.

Step 5: One bit will be denoted as acknowledgement bit to notify the master that slave is set to transmit or receive data.

Step 6: After the acknowledgement bit is accepted by the master it keeps on the information serially on the SDA line.

Step 7: The Initial IC sends alternately receives Concerning illustration 8-bit expression about information it needs. Following each 8-bit information expression the sending IC anticipates that the getting IC should recognize the exchange will be setting off correctly[6] .

Step 8: When each and every data is received STOP signal is produced and the bus can be free again.

Only two entities are considered in any communication i.e the Master which produces the signals and the Slave that responds when it is addressed.

3 TEST BENCH VERIFICATION

This Advanced Verification Methodology (AVM) offers a suitable structure to attain coverage driven verification (CDV).This CDV combines self checking test benches, automatic test generation, and coverage metrics to considerably reducing the time taken to verify a design. And ensure that thorough verification is done[7].This also eliminates the effort and time consumed for generating hundreds of tests.

Receive early warnings of faults and deploy error analysis to streamline debugging and runtime checking. The traditional directed testing flow is unique in relation to the CDV stream .Verification goals are also set firstly in CDV by means of controlled planning procedure. Then generate a test bench as in figure 3 that creates and sends legal stimulus to the DUT (I2C). For identification of undesired DUT behaviour, checkers are included. Simulations are carried out after both the coverage model and test

bench are implemented[8]. Verification is then attained.Detailed verification of the design is attained by changing the randomization seed or test bench parameters by Using CDV. Test constraints can be included on top of the framework so that verification goals are achieved rapidly.

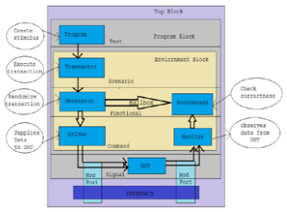

Fig 3 Verification Components of Test Bench

The following are the

1. transactions
2. Driver
3. Sequencer
4. Monitor
5. Agent
6. Environment

1.Transactions

Interfaces signifies the stimulus to the DUT. The aspects and fields of transactions would inferred from the transactions specification Previously, An test, a few information things would produced What's more the individuals would sent of the DUT(I2C) by means of driver. For the most part information thing fields are randomized utilizing framework Verilog imperatives a few amount for tests might make made.

Interface

Interface functions as a real link among the design-under-verification and the verification setting. It can be a System Verilog interface. Those interface characterizes those pin-level portrayal of the DUT. A interface may be essentially a pack about wires or nets.

In System Verilog, various modules would be joined through a module ports. For huge segments or complex designs this is not creative as physically joining hundreds or more of ports is not an simple job and this leads to error[9] .So thorough understanding of all the pins is needed. And on varying the design characteristics it is also needed to alter the pin association in the interface . But these difficulties would be overwhelmed by means of an interface. Interface will be used to link the DUT to other component or sub-modules.

The benefits can be considered as:

- Port descriptions are independent of modules
- Very simple to preserve the signals
- Well organized information stream between modules or blocks
- Interface can also have tasks and functions as well as protocol checking by means of assertions
- It is sent as a single data item.

Virtual Interface

Virtual interface offer a structure for differentiating abstract models from the original signals of the design.A virtual interface permits those same occurrence or the subprogram with work on distinctive parts of the plan. It vigorously controls those set about signs connected with those subprogram, this permits to pasquinade the same information things over all parts of the segments. Instead of stating to the original signals directly, we might work with a fixed number of virtual signals .

Sequence

A sequence a chain of transaction and also a sequencer can used to regulate the stream of transaction generation. A sequence is stretched from Avm_sequence class object. Avm_sequencer ensures the production of this series of transactions. Avm_driver receives the series of sequences from sequencer Furthermore techniques the packets of information things or sends them to the DUT or to the different part.

2.Driver

It is an active unit which follows logic that initiatives the DUT. The data items are constantly received by the driver and samples

them and drives it to the DUT. (If verification environment is produced in the past, driver functionality can be implemented) For example, a driver controls address bus, data bus and the read/write signal to DUT(I2C).

3. Sequencer

A Stimulus generator is sequencer that controls the data flow which are given to the driver for execution. In default case, a sequencer works similar to that of a simple stimulus generator and also returns a random data items on request from the driver. The default behaviour of driver permits to include limitations in data item class for monitoring the spreading of randomized values[10]. Transactors are used to randomize group of transactions while sequencer is used to capture key randomization requirements.

4.Monitor

It (Monitor) is a passive unit which samples Design under test(DUT) signals without driving them. Monitor gathers coverage information as well as perform checking. Data items are collected by a monitor and abstracts signal data from the bus and next transforms the data to a transaction which can be made accessible for supplementary modules and as well as to the test writer [11].

- Extracts events:A monitor figures out the accessibility of transaction, configurations of the data, and produces an event to inform the accessibility for other parts[12]. It also takes the status information so that it can be accessible to other modules and to the test writer.

- Accomplishes coverage and checking

5. Agent

Driver, Sequencer and monitor can be used individually. For reducing the amount of effort and knowledge as per the constraint of test writer, this methodology mentions that environment developers should generate more abstract model known as agent. Agents can verify DUT devices[13]. Verification Components contains more than one agent. Some agents initiate transactions to the DUT also, for example transmit agents or master , while other agents respond to transaction requests that is receive agents or slave. Agents must be configurable to be as either active or passive. Transactions are driven according to test instructions by Active agents . DUT activity is observed by passive agents.

6. Environment(env).

Environment is the The highest-level module of the Verification.It can have more than one agents, and a bus monitor[14]. The env contains config. properties which allows in customizing the topology and behaviour to mark it as reusable. For instance, active agents will be altered to passive agents when ever verification env is again used in system verification units[15].

4 RESULTS AND VERIFICATION

The I2C is carried out for the functional verification by means of Advanced Verification Methodology . The practical confirmation may be of the RTL outline of the I2C What's more produces the complete code coverage.

Coverage Type ◂	Bins	Hits	Misses	Coverage (%) ◂
Branch	98	97	1	98. 97%
Statement	45	45	0	100.00%
Toggle	107	91	16	85.04%
FSM	4	4	0	100.00%
Functional coverage				100.00%

Fig 4 :Coverage report of I2C

The verification method plays an vital role in the SOC design ,so it is performed in XILINX for RTL design and the testbench verification is performed in VCS. The RTL design is performed in HDL and the verification is performed in Advanced verification Methodology. Here I2C is used as DUT for the functionality verification and the coverage of code can be determined by means of VCS and obtained as 100 %.Different types of code coverage is carried out like Branch coverage ,Statement coverage, Toggle coverage and FSM coverage.

5 CONCLUSION

By means of AVM developing IP for any Design Under Test (DUT) becomes an easy task. I2C core based on Advanced Verification Methodology technique is verified by using VCS simulator.100% code coverage for RTL design is obtained. AVM gives the whole coverage of the RTL design in order to attain the error free design of I2C which would be executed practically.This can also be

further applied for the ASIC usage and System On Chip (SOC) applications.

References

[1] A. C. Guyton and J. E. Hall, Text Book of Medical Physiology, Elsevier. International Edition: 978-0-8089-2400-5.

[2] Hayat, T., Ali, N., and Asghar, S.,Hall effects on peristaltic flow of a Maxwell fluid in a porous medium, Phys. Lett. A 363,397-403, 2007.

[3] Lew, G.S., Fung, Y.C. & Lowensein, C.B. Peristaltic carrying and mixing of chime in small intestine. J.Biomechanics., 4, 297-315, 1971.

[4] Shukla, J.B. and Gupta, S.P. Peristaltic transport of a power-law fluid with variable viscosity. Trans.ASME. J. Biomech. Engg. 104, 182-186,1982.

[5] Vajravelu, K., Sreenadh, S. and Ramesh Babu, V., Peristaltic transport of a Hershel-Bulkley fluid in an inclined tube. Int. J. Non-linear Mech. 40, 83-90, 2005.

[6] Mishra, M., Ramachandra Rao, A.,Peristaltic transport of a power law fluid in a porous tube, J. Non- Newtonian Fluid Mech., 121, 163-74, 2004.

[7] Sreenadh, S., Lakshminarayana, P. and Sucharitha, G., Peristaltic flow of micropolar fluid in an asymmetric channel with permeable walls. Int. J. of Appl.Math and Mech., 7 (20), 18-37, 2011.

[8] Latham T.W. , Fluid motion in peristaltic pump, S M Thesis, MIT, Cambridge, MA, 1966.

[9] Shapiro A.H., M.Y. Jaffrin and S.L. Weinberg, Peristaltic pumping with longwave lengths at low Reynolds number, J. Fluid Mech. 37 799, 1969

[10] Hayat,T.,Noreen, S., Alsaedi, A., Peristaltic flow of pseudo-plasic fluid in an asymmetric channel. Journal of Applied Mechanics., 79, 054501-6, 2012.

[11] Vajravelu, K., Radhakrishnamacharya, G., and Radhakrishna murty, V., Peristaltic flow and heat transfer in a vertical porous annulus, with long wave length approximation, Int. J. Non-Linear Mech. 42, 754-759, 2007.

[12] Rathod, V,P,. and Laxmi, D., Effects of heat transfer on the peristaltic MHD flow of a Bingham fluid through a porous medium in a channel, International Journal of Biomathematics 7 (6), 1450060, 2014.