IMPLEMENTATION OF UVM TESTBENCH FOR VERIFICATION OF APB PROTOCOL

Sanjeeth Paul Gunza, MS in VLSI Engineering, VEDA IIT, Guntur, gsp97250@gmail.com

Sivalakshmi Devi Peram, MTS, Soctronics, Guntur, SivalakshmiDevi.Peram@invecas.com

SivajiSatyaGaneshNandigama, Assistant Professor, Dept of VLSI , VEDA IIT, Guntur, sivajisatyaganesh.nandigama@vedaiit.com

ABSTRACT

In verification of any design, test bench is required for applying the stimulus and getting back the result from the design. As signals required for the design increases, complexity in developing the traditional test bench and carrying out the verification process also increases. For complex designs, test cases are developed as per features of the design and they are applied through test bench. Implementation of traditional test benches for verification of complex designs also has certain drawbacks like lack of reusability and modularity and lack of control in stimulus generation. So, for complex designs like APB protocol UVM verification architecture for verifying the features of that protocol is proposed as it has advantages like modularity, reusability and it is also simulator independent[3].

INTRODUCTION

I. APB Protocol

APB protocol is one part in AMBA protocol family. This protocol actually provides low cost interface. It will also guarantee minimal power consumption. The low band width peripherals can be interfaced to this protocol[1]. It means that it is an un pipelined protocol[2]. In this all signals transitions taken into account only when it happens at rising edge of clock. For this protocol, every transfer takes two cycles[2].

Below is the table which represents the list of important signals that are used in this protocol and it's description along side

Signals	Description
PCLK	All transfers on this protocol are timed by this signal
PSELx	Slave device is selected by this signal
PENABLE	This represents that transfer has reached second cycle
PREADY	Signal indicates that slave is ready for transfer
PWRITE	Direction of the data transfer is indicated.
PWDATA	This will have data when PWRITE= 1.
PRDATA	This will have data when PWRITE=0

The below figure gives how this protocol operates.

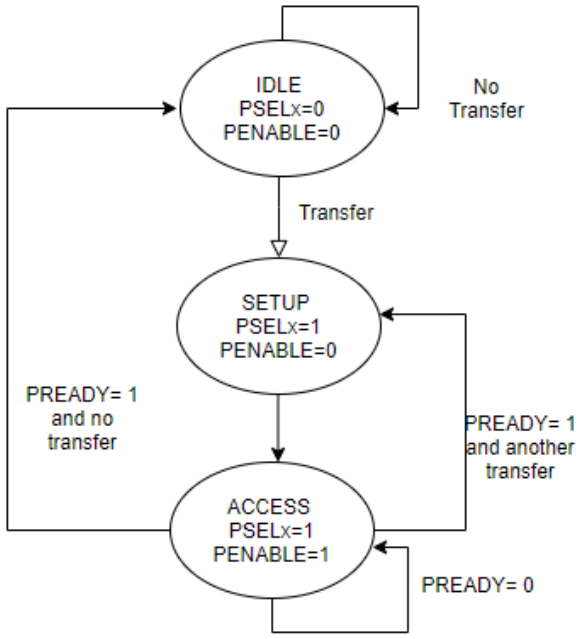

Fig 1 : Operational states in APB protocol

As per state diagram that has shown above, there are three operational states. They are IDLE, SETUP, ACCESS. In IDLE stage, transaction is at initial stage. If any data transfer or transaction has to be happen then thePSELx signal has to go to high. It represents SETUP stage in operation of APB. Bus has to remain only for one clock cycle here. After that PENABLE signal has to go to high . It represents ACCESS stage in APB operational states. In this stage PREADY signal plays a prominent role in deciding to which state the transfer will move on[2].

The transfer will not move from that phase until PREADY signal goes to high . It is also infer that the transfer will not complete until PREADY signal is asserted. It is also clear from the operating states that PSELx signal has to be enabled first followed by PENABLE signal and then PREADY signal. If the transfer is single transfer then the transfer will move to IDLE phase which means that PSELx and PENABLE signal has to de-asserted to zero[2].

II. UVM Testbench

In UVM Testbench, Design Under Test (DUT) and UVM Test class are instantiated and connections between them are configured

The description about the components present in the UVM Testbench are explained below

UVM Test : This is the top level component present in the Testbench. It is responsible for instantiating and configuring the top-level environment. It will also invoke sequences for applying the stimulus to DUT[3].

UVM Environment : This is responsible for instantiating other verification components that are inter related to each other . Components like UVM scoreboard and UVM Agent are instantiated under this.

UVM Scoreboard: This will carry out the checking of behavior of DUT . This component usually receives the output from UVM monitor as input and will compare with expected output.

UVM agent : In this other verification components are included that are dealing with DUT interface. In this component UVM Sequencer is responsible for stimulus flow and UVM Driver is responsible for applying stimulus on DUT and UVM monitor is responsible for collecting the output from DUT

UVM Sequencer : It is like an arbiter for controlling transaction flow from multiple stimulus sequences[3] .

UVM Driver : It will receive sequence-items from UVM sequencer and applies it on DUT interface. So driver has also to convert a transaction level stimulus into pin level stimulus for applying the data[3].

.

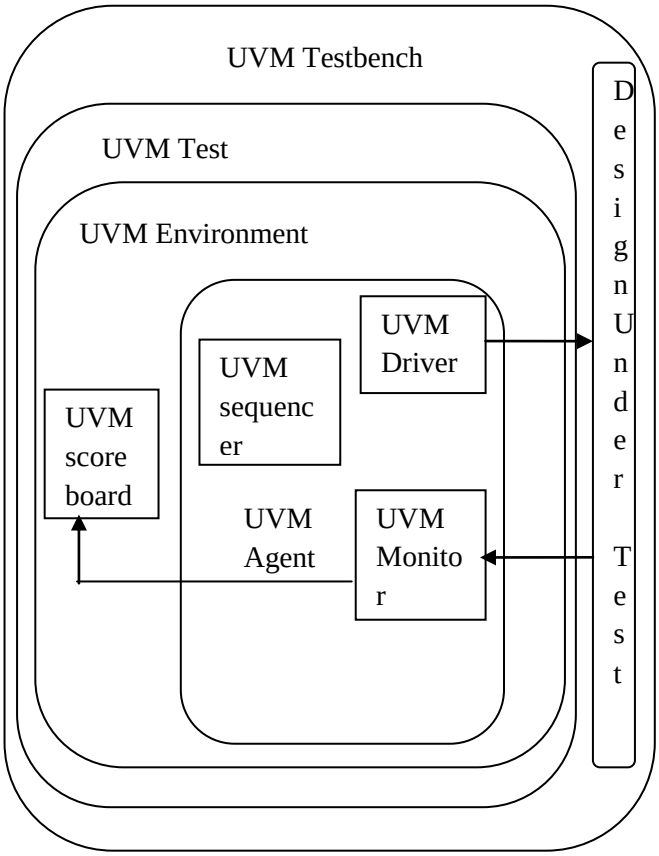

Fig 2 : UVM Architechture

UVM monitor : It will capture the output present at transaction in DUT interface and will also sent out the data to rest of UVM Test bench based on connections made . Monitor also has to convert the pin level stimulus to transaction level stimulus for carrying out the operations.

UVM Test bench is created by instantiating Design Under Test and UVM Test . Thereafter , UVM Test instantiates other UVM components and after that stimulus or sequences will be generated and will be applied through driver and output from the design is collected by monitor and will be passed to other UVM components as per connections[3].

Verification Scenarios

Scenario 1 :Writing data to design and reading back the data from design which is associated with APB protocol

TestDescription

Step 1 : At first positive edge of PCLK , PSELx signal along with PWRITE signal has to be asserted high and PWDATA and PADDR has to be given with required value.

Step 2 : After that at next positive edge of PCLK, PENABLE signal has to be asserted high.

Step 3 : At next positive edge of PCLK, PENABLE and PSELx and PWRITE has to be asserted low.

Step 4 : At next positive edge of PCLK,PSELx signal has to be asserted high and PWRITE signal has to be asserted low and PADDR has to be given with value that was given earlier in Step 1.

Step 5 : After that at next positive edge of PCLK, PENABLE signal has to be asserted high.

Step 6 : At next positive edge of clock, data will come from PRDATA . At nextpositive edge of clock , PSELx and PENABLE signal has to made low.

Step 7 : Data from PRDATA will be compared with PWDATA . If same test is pass else test is fail.

Result:

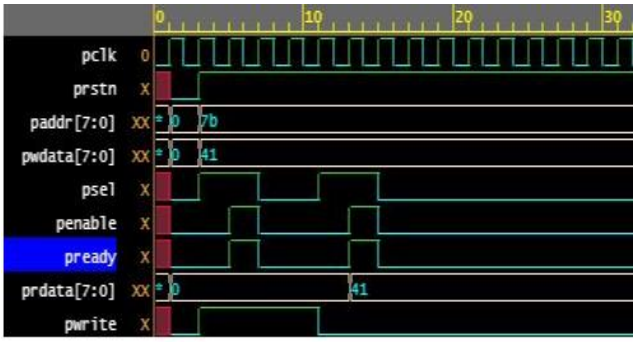

Fig 3 : Waveform of the result for scenario1

Scenario 2 : Whilewriting data to design , PENABLE signal has to be asserted low for two clock cycles after PSELx enabled and reading back the data from design which is associated with APB protocol. This is an negative scenario.

Test Description

Step 1 : At first positive edge of PCLK , PSELx signal along with PWRITE signal has to be asserted high and PWDATA and PADDR has to be given with required value.

Step 2 : After two consecutive positive edges of PCLK, PENABLE signal has to be asserted high.

Step 3 : At next positive edge of PCLK, PENABLE and PSELx and PWRITE has to be asserted low.

Step 4 : At next positive edge of PCLK,PSELx signal has to be asserted high and PWRITE signal has to be asserted low and PADDR has to be given with value that was given earlier in Step 1.

Step 5 : After that at next positive edge of PCLK, PENABLE signal has to be asserted high.

Step 6 : At next positive edge of clock, data will come from PRDATA . At next positive edge of clock , PSELx and PENABLE signal has to made low.

Step 7 : Data from PRDATA will be compared with PWDATA . If same test is pass else test is fail.

Result:

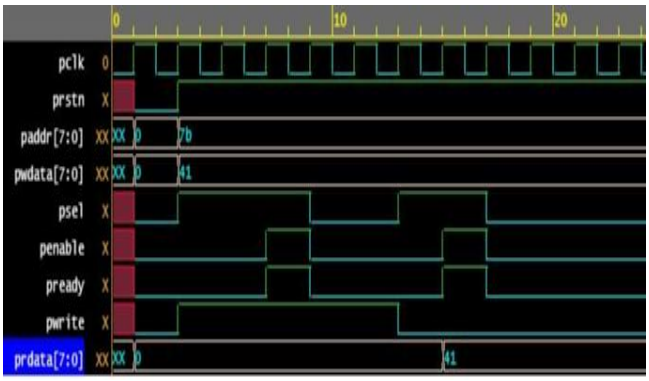

Fig 4 : Waveform of the result for scenario 2

Scenario 3 : In this scenario, at first positive edge of clock sel signal and write signal are asserted along with address and data . At next edge of clock the address value is changed with the assertion of enable signal. After the write transfer completed, Read transfer will be done

from the address that has given initially and the changed address . As per protocol, address should not be changed after it is given with sel signal. This scenario is to make sure that even though address is changed design can take the address that is given with sel signal and should ignore the changed address and also developed because the protocol is widely used in programming peripheral registers. In such cases , data may written in wrong locations and it will cause further complexions.

Test Description :

Step 1 : At the positive edge of clock , PSELx and PWRITE signal has to be asserted and PADDR and PWDATA has to be given with values.

Step 2 : At the next positive edge of clock, PENABLE signal has to be asserted and PADDR has to be given with other value.

Step 3 : After that positive edge of clock, PSELx signal and PWRITE signal has to be deasserted along with PENABLE signal.

Step 4 : After two positive edges of clock, PSELx signal has to be asserted again and PWRITE signal has to keep low and PADDR has to be given with the value that has given in step 1

Step 5 : At the next positive edge of clock, PENABLE signal has to be asserted and data from PRDATA has to be taken and has to compare with data given in PWDATA. If same test pass else test fail.

Step 6 : After two positive edges of clock, PSELx signal has to be asserted again and PWRITE signal has to keep low and PADDR has to be given with the value that has given in step 2

Step 7 : At the next positive edge of clock, PENABLE signal has to be asserted and data from PRDATA has to be taken and has to compare with data given in PWDATA. If different test pass else test fail.

Result :

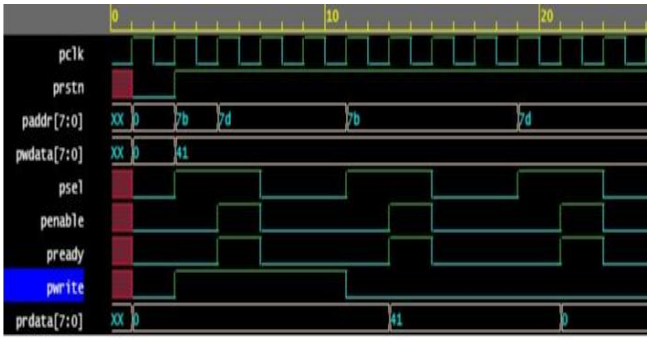

Fig 5 : Waveform of the result for Scenario 3

Scenario 4 : In this scenario, while write transfer is going on reset is applied with value x . This action is taken into consideration to check whether the reset with x is effecting the design or not . This will be confirmed further by doing the read transfer for the same address. If the value is same as the value that it is written , It can be confirmed that the reset given with value x does not affect the design.

Test Description:

Step 1 : At first positive edge of PCLK , PSELx signal along with PWRITE signal has to be asserted high and PWDATA and PADDR has to be given with required value.

Step 2 : After that at next positive edge of PCLK, PENABLE signal has to be asserted high and PRSTn signal has to be given with value x.

Step 3 : At next positive edge of PCLK, PENABLE and PSELx and PWRITE has to be asserted low and PRSTn signal has to be given with value 1.

Step 4 : At next positive edge of PCLK, PSELx signal has to be asserted high and PWRITE signal has to be asserted low and PADDR has to be given with value that was given earlier in Step 1.

Step 5 : After that at next positive edge of PCLK, PENABLE signal has to be asserted high

Step 6 : At next positive edge of clock, data will come from PRDATA . At next positive edge of clock , PSELx and PENABLE signal has to made low.

Step 7 : Data from PRDATA will be compared with PWDATA . If same test is pass else test is fail.

Result :

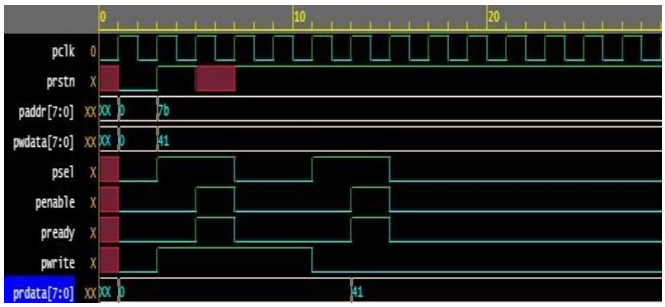

Fig 6 : Waveform of the result for Scenario 4

Conclusion

In this paper, verification of APB protocol is done with UVM test bench . Verification is done for positive scenario and also done for negative scenario. Verification for negative scenarios are also required because for checking the robustness of RTL these scenarios are also required.

References

1."Design and verification of AMBA APB protocol" , KommirisettiBheemaRaju, Bala Krishna Konda, IJERT, January 2017, Vol-7, Issue-1,pp 87-90.

2."ARM AMBA 3 APB Protocol v1.0 Specification", ARM Limited

3."UVM 1.2 user guide", Accellera Systems Initiative