International Journal of Engineering Research and General Science Volume 2, Issue 4, June-July, 2014
ISSN 2091-2730

AXI Interconnect Between Four Master and Four Slave Interfaces

Mayank Rai Nigam[1], Mrs Shivangi Bande[2]

[1] Scholar, IET DAVV Indore

[2] Associate Professor, IET DAVV Indore

E-mail- kayasthmayank19@gmail.com

Abstract— The ARM (Advanced RISC Machine) has developed AMBA (*Advanced Microcontroller Bus Architecture*) bus protocol which is widely used by System-on-Chip (SoC) designers. Systems-on-Chip are one of the biggest challenges engineers ever faced which result a mix of microprocessor, memories, buses architectures, communication standards, protocol and interfaces.AMBA buses act as the high-performance system *backbone* bus. It supports the efficient connection of processors, on-chip memories and off-chip external memory interfaces. APB and AHB come under AMBA standard. ARM has come up with its latest on chip bus transfer bus protocol, called *AMBA AXI*. AXI stands for *Advanced Xtensible Interface*.From a technology perspective, AMBA AXI (Advanced eXtensible Interface) provides the means to perform low latency, high bandwidth on chip communication between multiple masters and multiple slaves. Moving one stage further, from an implementationperspective, configurability and programmability are becoming vital to ensuring IP can be tuned for a given application or project requirement.

Keywords: vhdl,fpga,digital design,protocol,axi,Xilinx,channel etc.

Introduction

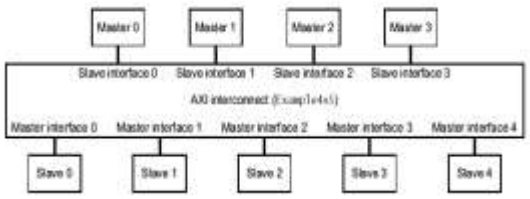
Interconnect provides efficient connection between master **(***e*.g. ARM processors, *Direct Memory Access* (DMA) or *Digital Signal Processor* (DSP)**)** and slave **(**e.g. external memory interface, APB bridge and any internal memory**).**

The Interconnect is a highly configurable RTL component, which provides the entire infrastructure require to connect number of AXI masters to a number of AXI slaves. This infrastructure is an integral part of an AXI-based system.

Architecture of interconnect is highly modular with each of the routers and associated control logic partitioned on a per-channel basis. It ensures, which bus master is allowed to initiate data transfers depending on *highest priority* or *fair* access.

As AXI provides many features such as out of order completion, interleaving; interconnect is responsible to take care of interleaving and out of order. The block level RTL code is automatically configured from a system description file to specify no of master, slave , width of address bus hence interconnect is implemented depending on the application requirements.

AXI Interconnect takes care of all 5 channels, using which data transfer between master and slave take place.



Example of AXI Interconnect

Features of Interconnect

The ACI features are:

- It is compliant with the *AMBA AXI Protocol v1.0 Specification*
- It multiplexes and demultiplexes data and control information between connected masters and slaves
- It enforces the AXI ordering rules that govern the flow of data and control information on different channels
- It has a multi-layer capability to allow multiple masters to access different slaves simultaneously
- Out-of-order data support
- You can configure the following parameters:

— number of master and slave interfaces

— The ID width of each slave interface

— The read and write acceptance capability of each slave interface

— The write issuing capability of each master interface

— The write interleave capability of each master interface

AIM OF THE PROJECT

The aim of the project is to Design an AXI interconnect between four master and four slave interfaces.

OBJECTIVE

Design related TASKS

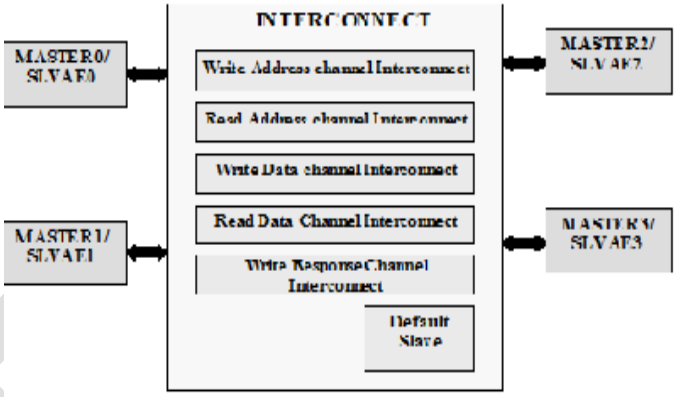
Design related tasks that were performed in the project are:

- The Architecture of the design was thought by considering the specifications and then Block Diagram was prepared.
- The Block Diagram was divided into sub- modules which are communicating with each other.
- Block Diagram of 5 channels are made
  - Write Address Channel
  - Read Address Channel
  - Write Data Channel
  - Read Data channel
  - Write Response Channel
- Block diagram was analyzed number of times for the correctness of the architecture
- After the designing ,Verilog and VHDL coding is done of low level modules used in all the channels
- These low level modules are combined in a top module for all the block diagram of the 5 channels
- All codes corresponding to these block diagrams have been combined in top level module which constitutes the whole interconnect
- Whole Design was synthesized to check for the errors.

SPECIFICATIONS

- Design the AMBA AXI INERCONNECT for four Ports in which each port behave as AXI based master interfaces and slave interfaces.
- 32 Bit Address Bus and 64 Bit Data Bus
- Configurable Port Addresses (Slave size is configurable)
- One outstanding transaction
- Support all type of Burst Transaction (Wrap, INCR, Fixed)
- Support Normal and Locked Operation
- Support 200 MHz on VIRTEX 5
- Following is the priority considered for masters :
  Master0 > Master1 > Master2 > Master3

BLOCK DIAGRAM AND DESCRIPTION



- **Master** generates and drives transaction onto the bus.
- **Slave** device accepts transaction from any master.
- **Interconnect** routes the AXI requests and responses between AXI masters and AXI slaves. Passive Monitoring, Checking and Collection of functional coverage specifically targeted at the AXI Interconnect are the mains functions of Interconnect.

Interconnect consist of 5 channels:

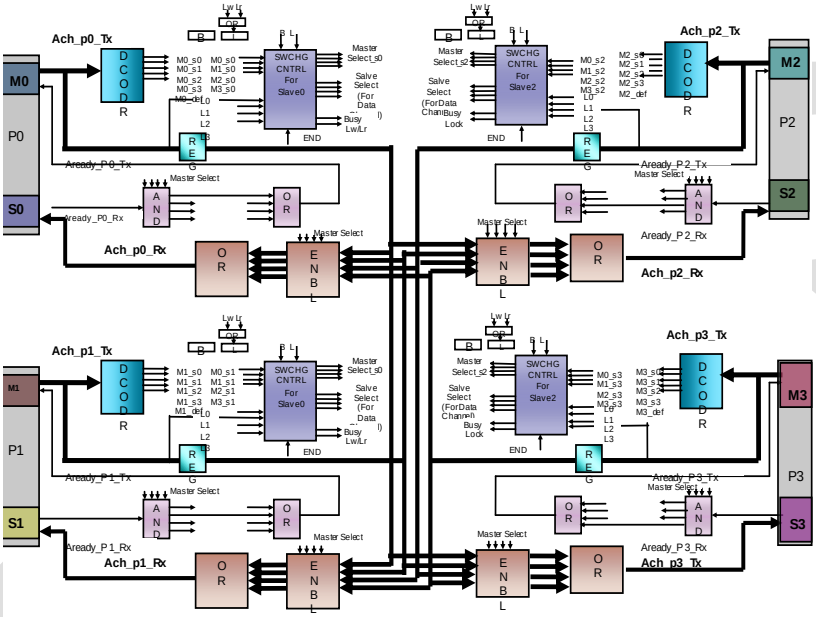
- **Read address channel:** This channel gives information about Transaction ID for read operation, address of slave, Burst length along with size and type, valid signal to indicate control information is valid and ready.
- **Write address channel:** This channel gives information about Transaction ID for write operation, address of slave, Burst length along with size and type, valid signal to indicate control information is valid and ready
- **Read data channel:** This channel gives information about Transaction ID for read data, read data, read response along with ready and valid signal
- **Write data channel:** This channel gives information about Transaction ID for write data, write data with strobe information ready and valid signal
- **Write response channel:** This channel gives information about Transaction ID for write data, write response along with ready and valid signal
- **Default slave** is used when there is no fully –decoded address map physically present. There can be address at which there are no slave to respond to the transaction, then interconnect effectively routes the access to a default slave. As in case of AXI protocol it is necessary that all transaction must be complete even there is any error.

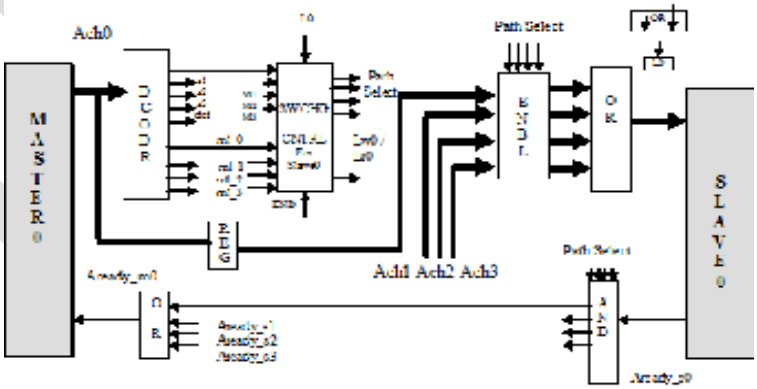
ADDRESS CHANNEL

The address channel conveys address information along with control signal from master to the slave. AXI support different address buses for write and read transfer, so that through put of the system is increased. Both channels (read/write) have same signals included in the address channel.

The address channel includes ***address bus*** which is 32 bit, ***length*** of burst; it gives exact no of data transfer in burst, ***size*** of transfer to indicate bytes in one burst, ***burst*** type which is WRAP, FIXED and INCR, ***lock*** information along with ***valid*** and ***ready*** signals

Block diagram of address channel





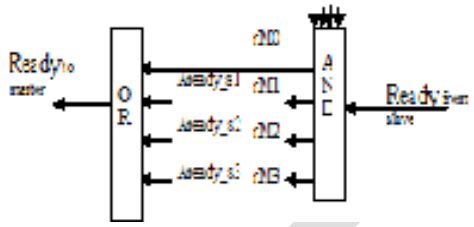
one master one slave address channel

Following points explain the detailed functioning of Address channel interconnects

1. When Master sends valid Address and control signal *Slave Decoder* decodes that address and generate output to indicate request from master to slave.
2. Decoder has five output bits, each bit indicates request to particular slave S0, S1, S2, S3 and default slave.
3. Each decoder output is given to the each *switching control unit* as request.
4. S0 is given to *switching control unit* for slave0; S1 is to *switching control unit* for slave1 and so on.
5. Thus each *switching control unit* receives four request from four masters It gets request from each master for NORMAL and LOCKED operation, depending on priority it will grant that slave to appropriate master.

- *Path select* will enable granted master address channel other channel will remain disabled.
- If the select signal is 1000 then master0 address channel is selected
- If the select signal is 0100 then master1 address channel is selected
- If the select signal is 0010 then master 2 address channel is selected
- If the select signal is 0001 then master 3 address channel is selected

6. Slave will accept now valid address and control signals.
7. Slave sends ready signal back to the granted master. This ready signal is given to granted master by ANDing logic. This is done in same way as address and control signals are routed towards the slave.
   Thus Master receives ready from slave.



- If the path select signal is 1000 then rM0 will be set.
- If the path select signal is 0100 then rM1 will be set.
- If the path select signal is 0010 then rM2 will be set.
- If the path select signal is 0001 then rM3 will be set.
- rMx indicates ready signal goes to master x (x = 0,1,2,3)
  Master side ready signal is received by OR logic. All signals (Aready_S0, Aready_S1, Aready_S2, Aready_S3) coming to OR is from each slave. Design assures that out of all incoming signals to OR logic, at a time only one will be set.

8. After receiving ready signal master de-asserts valid signal of address channel.
9. This *switching control* won't accept any further request for that slave till completion of transaction.
10. *Switching control unit's* output will remain in same state till *End* signal is received. Which indicates transaction is completed.

For operation of address channel let us understand operation of each block in the channel with detail. Following points explain the detailed functioning of Address channel interconnect

❖ ***DECODER:***

- Decoder functions as an address decoder which generates control signals to enable or disable the channels to a particular slave from particular master.
- Decoder can receive the valid request (ARADDR [32] / AWADDR [32]) for read or write operation from any of the four masters.
- The decoder decodes the address by comparison to memory maps of slaves and generate control signals to enable master request to the appropriate slave.

If the ***start address*** is 00000000 hex and ***end address*** is 00000fff then control signal enable the channel for S0 slave

If the ***start address*** is 00001000 hex and ***end address*** is 00001fff then control signal enable the channel for S1 slave

If the ***start address*** is 00002000 hex and ***end address*** is 00002fff then control signal enable the channel for S2 slave

If the ***start address*** is 00003000 hex and ***end address*** is 00003fff then control signal enable the channel for S3 slave.

If M0 wants to send the valid address and control information to the slave0 then Master0 will generate the address which lies in the starting and end address of slave 0. Output of the decoder which is 5 bit signal will generate slave0 pin active high that is 1 and the rest of the bits for slave 1, slave 2 , slave3 and default slave is low that is zero .

The active high signal for slave 0 is connected to the switching control unit of slave 0.

Switching control

Switching control Description:

- Switching control accepts requests form all *four* masters for normal or locked operation.M0L0 bits are for *normal* and *locked* operation request from *master0*.
  Similarly M1L1 bits are from *master1*, and so on.

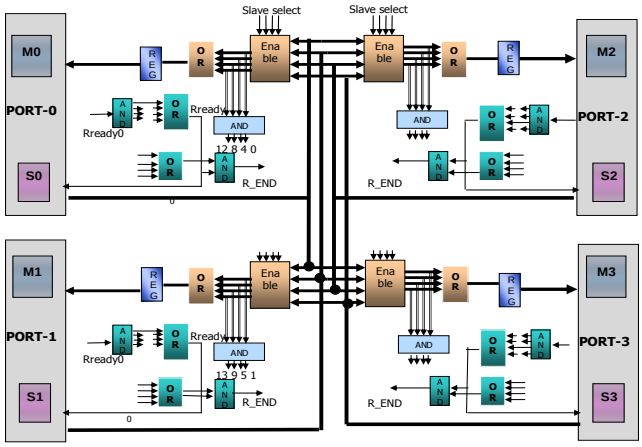
- Mx is active high bit, indicates request for slave from *master x*. It out put of slave decoder.
- Lx bit indicates *normal* operation if it is '0' else *locked* operation.
- Other inputs to *switching control unit* are Busy, Lock and End signal, Busy and Lock signals shows status of slave whether it's being accessed by other master.
- End signal brings *switching control unit* to idle state on end of transaction.
- Master select outputs used to select channels coming from mater to slave. I.e. address and write data channels.
- Slave select outputs used to select channels coming from slave to master. i.e. Read data channel and write response channels.

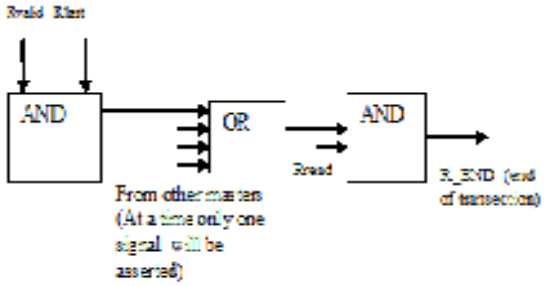
READ DATA CHANNEL----------

The read data channel conveys both the read data and read response information from the slave back to the master. The read data channel includes:

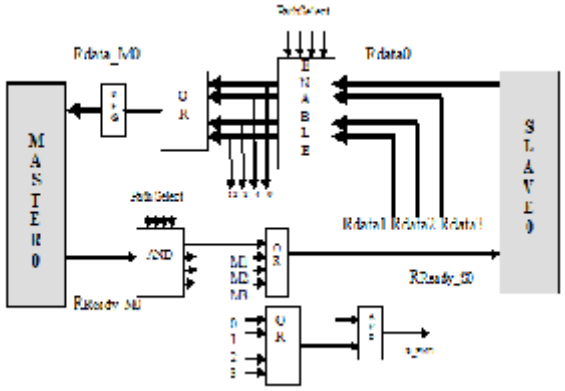
data bus, which can be 8, 16, 32, 64, 128, 256, 512, or 1024 bits wide and read response indicating the completion status of the read transaction.

Block diagram of Read data channel





In read operation slave will send valid read data, this data is routed by switching control unit's output. Following points explain the detailed functioning of read data channel interconnect



1. The process starts when master sends an address and control information on the read address channel along with valid information.
2. This address is decoded first to find which slave is to be access. Now signal will be given to the switching control logic of particular slave. It generates appropriate enable signal to select particular master's path to that slave.
   In the above case if select signal for slave 0 is generated by the arbiter then this select signal select particular master to read the data from the slave 0.

   - If the select signal is 1000 then data is given to master0
   - If the select signal is 0100 then data is given to master1
   - If the select signal is 0010 then data is given to master 2
   - If the select signal is 0001 then data is given to master 3
   - In above case if select signal is 1000 then read data to master 0 is selected from slave 0.
   - From this stage read data channel come into the picture. Data from all the four slaves (Rdata_S0, Rdata_S1, Rdata_S2, Rdata_S3) may available at the master ENABLE block, this enable block will select only that slave which is to be connected to the particular master.

3. Enable module blocks the data to be unintentionally passing to the master from slave. As master has not given any request no slave is selected and contains on the data bus is zero.
4. When master will assert Ready_M0 signal on bus at that time data from slave0 is accepted by master. This Ready_M0 signal is first given to the AND block which will assert only that signal which is going to the slave0. At slave0 Ready_M1, Ready_M2, Ready_M3 are also connected but as project support only one out standing transaction at a time only one READY signal is high.
5. Slave internally calculates next address from the address specified by the master in the address channel. Data on that address location is put by the slave on to the data bus along with valid signal to indicate that, there is valid data present.
6. Master will accept data when he will assert Ready_M0 signal high.

7. This process proceeds until final transfer in the burst takes place. At the final transfer slave asserts the RLAST signal to show that last data item is being transferred.

❖ **R_END signal generator block**

When RLAST signal appears on the line from slave along with RVALID and RREADY signal from same master, are used to generate the R_END signal. This signal is given to the switching machine which will then reset all previous output set by switching control logic block, as all Read data burst is transferred from slave to the master.

In the above case if slave0 is transferring data to master0 then path select will be "1000" to enable slave0's data path.

At the RLAST signal from slave0 signal '0' will be active high which is OR with RVALID and result of this is AND with RREADY signal of master0.

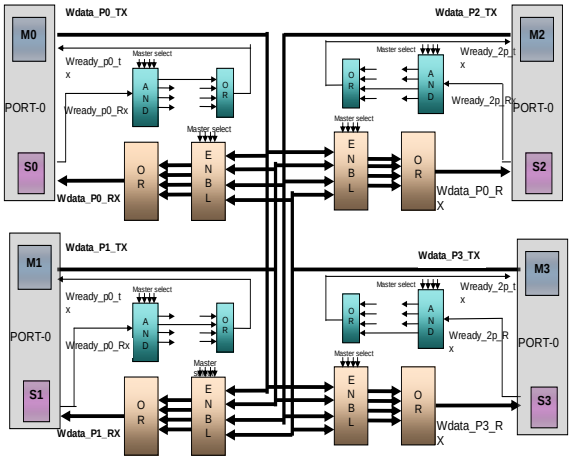
WRITE DATA CHANNEL----------

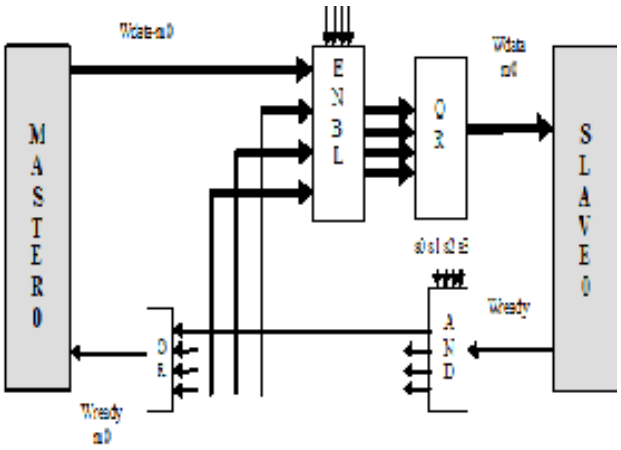
The write data channel conveys write data information from the master to the slave. The write data channel includes data bus, which can be 8, 16, 32, 64, 128, 256, 512, or 1024 bits wide and strobe indicating the valid byte lanes.

- During the write burst, master can assert the WVALID signal only when it drives valid write data .WVALID must remain asserted until the slave accepts the write data and asserts WREADY signal.
- **Write Strobe** Signal, WSTRB enables data transfer on the write data bus. Each write strobe signal corresponds to one byte of the write data bus. When asserted, a write strobe indicates that the corresponding byte lane of the data bus contains valid information to be updated in memory. There is one write strobe for each eight bits of the write data bus.

A master must ensure that the write strobes are asserted only for byte lanes that can contain valid data as determined by the control information for the transaction.

Block diagram of Write data channel





Following points explain the detailed functioning of write data channel interconnect:

- In Write Data channel the master will write data to the slave.
- The process starts when master sends an address and control information on the write address channel along with valid information.
- This address is decoded to find slave no. now signal will be given to the switching control logic for particular slave. It generates appropriate signal to enable particular master's path to that slave. The select signal which is 4 bit is generated from the SWITCHING CONTROL BLOCK. According to this select signal particular master and slave is selected in order to write the data.

In the above case if select signal for slave 0 is generated by the arbiter then this select signal select particular master to write the data to the slave 0.

- If the select signal is 1000 then master0 data is selected
- If the select signal is 0100 then master1 data is selected
- If the select signal is 0010 then master 2 data is selected
- If the select signal is 0001 then master 3 data is selected

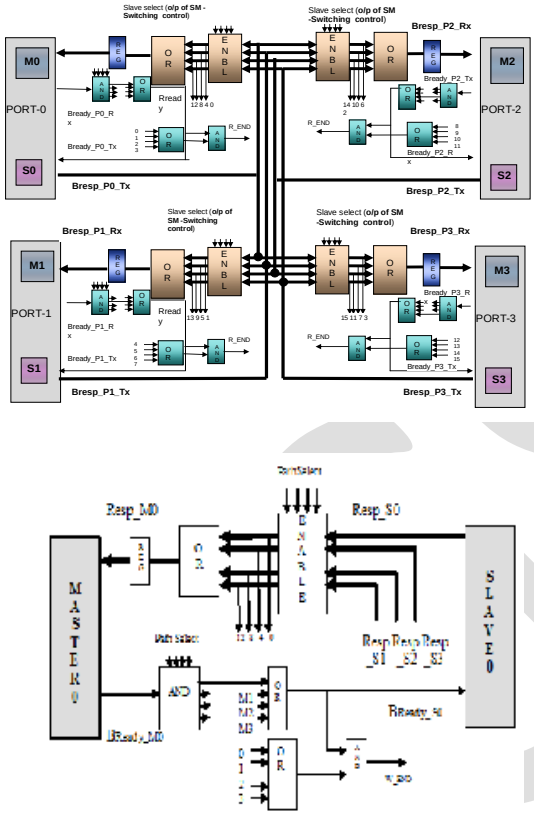
In above case if select signal is 1000 then write data of master 0 is selected and moves to slave 0

- From this stage write data channel come into the picture. Data from all the four masters (Wdata m0, Wdata m1, Wdata m2, Wdata m3) is available at the slave ENABLE block, this enable block will select which master is going to write particular slave.
- At the same time slave0, slave1, slave2, slave3 will send WREADY signal to the master from the AND block.
- This AND block also consist of select signal which selects to which master the WREADY signal is send. If the select signal is 1000 then master0 is selected to send Wready signal
- As soon as master0 get the WREADY signal master sends the next data to the slave.
- At the end of transfer LAST signals will be sending by the master indicating the end of transaction.

WRITE RESPONSE CHANNEL

The write response channel provides a way for the slave to respond to write transactions. All write transactions use completion signaling. The completion signal occurs once for each burst, not for each individual data transfer within the burst.Response channel is mainly used to indicate status of the write transaction. In case of write data transfer all data is coming from master side and slave does not acknowledge any thing to the master. Hence response channel is combined with the write data channel for acknowledgement from slave side.

Most important signal of Write response channel is BRESP. This signal is of 2 bit and indicates status, such as OKAY, EXOKAY, SLVERR and DECERR.



Response channel is used for acknowledgement. Slave can assert signals on this channel to indicate status of transfer. Design of this channel is same as that of read data channel (as both channel transfer data from slave to master) only signals are different.

In this operation slave will send response signal, which is routed by switching control unit's output. Following points explain the detailed functioning of response channel interconnect

1. When master sends an address and control information for write transfer after transferring all data response is send by the slave i.e. after WLAST signal from master side.
2. When address is send then decoder in address channel will select slave and send request to switching control of particular slave. Output generated by the switching control, are hold until response channel does not give W_END signal.

If master0 want to access slave0 then select signal for slave 0 is generated by the arbiter. This select signal selects particular slave for response of write transaction.

- For select signal 1000 Response is given to master0
- For select signal 0100 : Response is given to master1
- For select signal 0010 : Response is given to master2
- For select signal 0001 : Response is given to master3

In above case if select signal is 1000 then read data to master 0 is selected from slave 0.

3. Response channel signal from all the four slaves (Rresp_S0, Rresp_S1, Rresp_S2, Rresp_S3) may available at the master ENABLE block, this enable block will select only that slave which is to be connected to the particular master.
4. Enable module blocks the response of other channel to be unintentionally passing to the master from slave. If master has not given any request then no slave path going toward master is selected and output of this block is zero.
5. When master will assert BReady_M0 signal on bus at that time response signal from slave0 is accepted by master0. This BReady_M0 signal is first given to the AND block which will assert only that signal which is going to the slave0. At slave0

BReady_M1, BReady_M2, BReady_M3 are also connected but as project support only one out standing transaction at a time only one BREADY signal will be high.

6. BVALID must remain asserted until the master accepts the write response and asserts BREADY. The default value of BREADY can be HIGH, but only if the master can always accept

W_END signal generator block

BVALID and BREADY signal are used to generate the W_END signal. This signal is given to the switching machine which will then reset all previous output set by switching control logic block, as all Write data burst is transferred from master to the slave.

In the above case if slave0 is transferring response to master0 then path select will be "1000" to enable slave0's response path.

BVALID signal from slave0 i.e. signal '0' will be active high which is AND with BREADY signal of master0.

DEFAULT SLAVE

With 5 channels, another important block of AXI interconnect is *Default slave*. When interconnect cannot successfully decode a slave access (i.e. when slave is not present at Physical location specified by the master), it effectively routes the access to a default slave, and the default slave returns the DECERR response.

Fig. shows waveform for write data transaction. Master sends address first, READY signal indicate that master can send write data. After receiving last data from master slave gives response to indicate status of transfer.

The AXI protocol responses are:

- **OKAY**
- **EXOKAY**
- **SLVERR**
- **DECERR**

Decode error is generated typically by an interconnect component to indicate that there is no slave at the transaction address. AXI protocol requires that, all data transfers in a transaction should be completed, even if an error condition occurs. As one's the master places an address, it keeps on waiting until the address is not accepted. So someone has to accept this invalid address and complete the burst corresponding to it.

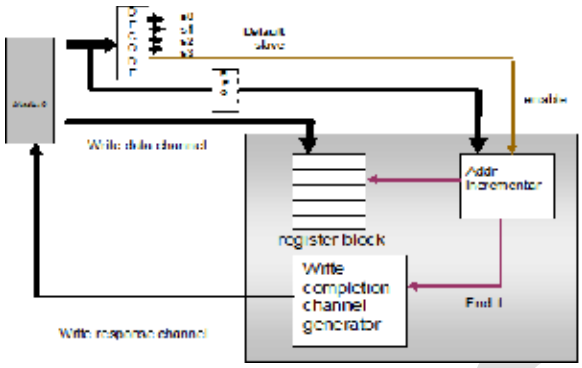
Therefore any component giving a DECERR response must meet all requirements to complete data transfer and generate appropriate signal along with response signal "DECERR".

This is where DEFAULT SLAVE comes into picture. Default slave will accept such invalid addresses and will complete transactions corresponding to such addresses by responding with a special error in response called "DECERR", which means decoding error. This error is meant to tell the master that no device is having the address for which transaction has been requested.

So default slave will be having two sections. One of these sections will handle write transactions and other will handle read transactions.

Default slave write section:

The DECODER in write address channel interconnect enables default slave and routes the invalid addresses along with control information attached with them to the DEFAULT Slave's write section.



Block Diagram of Default Slave for write transaction

Also the write data corresponding to these invalid transactions is accepted by default slave, as soon as the LAST data arrives, Default slave places a write response corresponding to this transaction on the write response channel. It also gives ready and BID signal to fulfill the protocol requirement.

In this way as specified in AXI specification, even the invalid transaction is completed by the default slave.

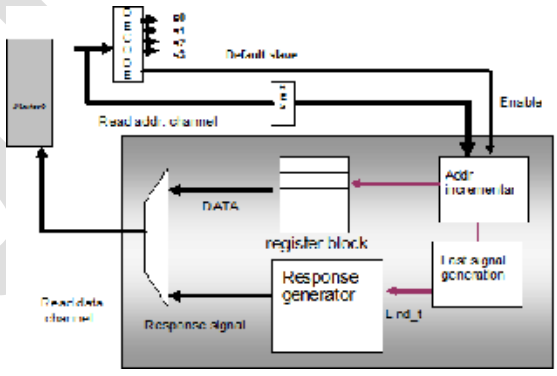
The block diagram of default slave's write section and its functioning are explained in following sections.

Following points explain the working of write section of default slave:

The DECODER in write address channel interconnect enables default slave. After enable signal default slave will assert AWREADY signal.

- AWID, AWLEN, AWSIZE, AWBURST are taken into account when AWVALID is high. AWID is used to generate BID signal.
- After accepting all control information default slave will write location by WDATA.
- As soon as WLAST signal is received default slave will enable write completion channel generator block and this block will generate appropriate error signal on BRESP bus.

Default slave read section:



Block Diagram of Default Slave for read transaction

This section, also work in the same way; as write section works. The DECODER in read address channel interconnect enables default slave and routes the invalid addresses along with control information attached with them to the DEFAULT Slave's read section.

Now default slave will give data to master by reading the information from register. Along with the LAST data, Default slave places a response corresponding to this transaction on the channel.

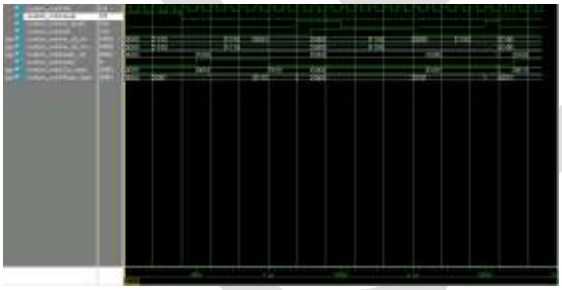
In this way as specified in AXI specification, even the invalid transaction is completed by the default slave.

Following points explain the working of write section of default slave:

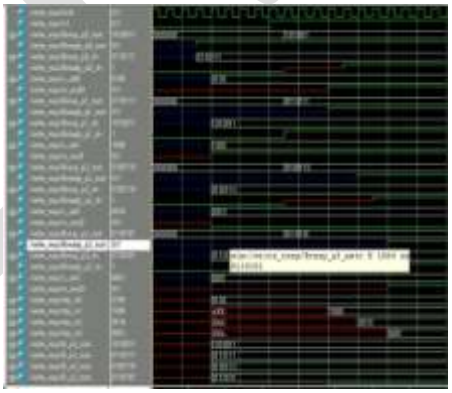
- The DECODER in write address channel interconnect enables default slave. After enable signal default slave will assert ARREADY signal.
- ARID, ARLEN, ARSIZE, ARBURST are taken into account when ARVALID is high. ARID is used to generate RID signal.
- After accepting all control information default slave will read location and send data on RDATA.
- It will calculate total burst size by considering ARLEN, ARSIZE and ARBURST signals. This value is decremented.
- As soon as burst size value goes to zero (i.e. End_t signal is generated) default slave will assert RLAST signal along with error signal on RRESP bus.

simulation results

Simulation result for the decoder

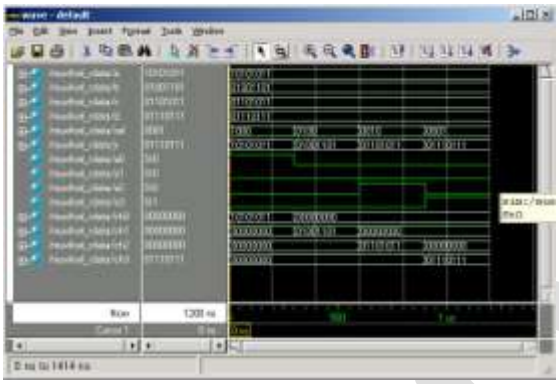


Simulation result for the Switching



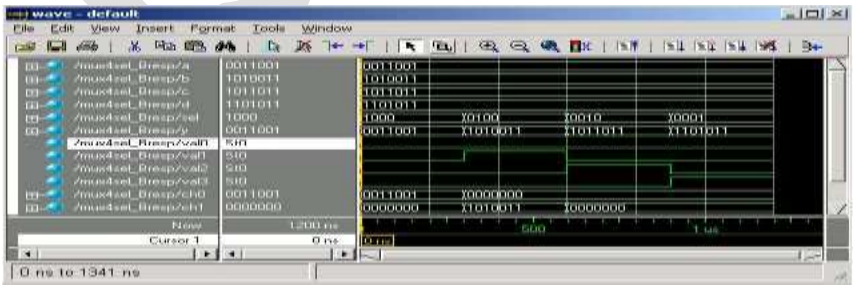
Simulation result for the Write Response Channel

Simulation result for the enable



Simulation result of Mux select for read data channel

simulation result for mux select



simulation result of Mux select for write response channel

Conclusion

- Functional verification is achieved successfully.
- The interconnect can works at 100 MHz frequency at Vertex E as target Device. (synthesized by xilinx ISE 8.2i).

All the Errors and warnings were removed successfully from design coding except one warning "signal is assigned but never use

REFERENCES:

Sr. No	Title of book	Author name
1.	***AMBA AXI protocol v1.o specification***	ARM Limited.
2.	***PrimeCell AXI Configurable Interconnect (PL300)*** Technical Reference Manual	ARM Limited.
3.	***AMBA Design Kit*** Technical Reference Manual	ARM Limited.
4.	***VHDL primer***	j.bhasker
5.	Fundamentals of Digital 0-07-116168-6 Logic with VHDL Design.- McGraw-Hill, 2000.	Stephen Brown, Zvonko Vranesic
6.	The Designer's Guide to 1-55860-674-2 VHDL(2nd Edition).- Morgan Kaufmann	Peter J.Ashenden
7.	VHDL(3rd Edition).- 0-07-049436-3 McGraw-Hill	Douglas L. Perry