₹ 200 ISSN - 2249-555X
Volume : 1 Issue : 8 May 2012

Journal for All Subjects

www.ijar.in

Listed in International ISSN Directory, Paris.

ISSN - 2249-555X

Indian Journal of Applied Research

Journal for All Subjects

Editor,
Indian Journal Of Applied Research
8-A, Banans, Opp. SLU Girls College, New Congres Bhavan, Paldi,
Ahmedabad-380006, Gujarat, INDIA
Contact.: +91-9824097643 E-mail : editor@ijar.in

INDEX

Volume : 1 | Issue : 7 | May 2012 ISSN - 2249-555X

Research Paper Engineering

Implementation of Self controlled Arbiter for High Speed Communication in on-chip

* Kaushik Mukherjee ** A.Ch. Sudhir *** Dr. B Prabhakara Raob

*, ** ECE Dept, GIT, GITAM University, Visakhapatnam

*** ECE Dept, JNTU, Kakinada

ABSTRACT

In this paper, the high performance bus matrix as an interconnection scheme has been developed, which supports slave side arbitration scheme rather than master side arbitration scheme. These two can be differentiated in terms of their request and grant signals. In the slave side arbitration scheme, master merely starts a burst transaction and wait for the slave response to proceed to the next transfer. However the multi-layer bus matrix of ARM offers only transfer based fixed and round-robin arbitration schemes. But in this paper, the arbiter supports three priority policies- Fixed, Round Robin and Desired transfer Length. And also supports three data multiplexing modes- transfer, transaction and desired transfer length. So there are nine possible arbitration schemes, out of which one will be selected by the arbiter at any time instant, depending upon the priority level and desired transfer length. So this arbiter is self motivated, and its modified structure given in this paper is quite simple rather than the previous one.

Since this newly modified arbiter structure is quite simple, therefore number of gates in the gate level net-list is less, in comparison to the previous one. Therefore it causes less delay in the circuit. It can be experimentally proved that, this arbiter will cause much faster communication and overall throughput is maximized. Since it gives the benefit of both increased Bandwidth and more flexible system structure, it will be useful in many SoC design, IP design, SRAM design as well as processor design.

Keywords : Multi Layer AHB, on-chip bus, SM arbitration scheme, Slave side arbitration scheme, System on chip (SoC)

INTRODUCTION

As CMOS technology evolves and applications became more complex with increased levels of hardware and software sharing (communication, multimedia, video games, networking), SoC design require a system bus with high bandwidth to perform multiple operation in parallel. To solve the Bandwidth problems, there are several types of high-performance on-chip buses proposed, such as multi-layer AHB (ML-AHB) bus matrix from ARM, the PLB cross bas switch from IBM, and CONMAX from Silicore. Among them, the ML-AHB bus matrix is widely used in SoC design due to its simplicity.

The ML-AHB bus matrix is an interconnection scheme based on AMBA AHB protocol, which enables parallel access path between multiple masters and slaves in a system. This is achieved by using more complex interconnection matrix and gives the benefit of both increased overall bandwidth and more flexible system architecture.

The unit of arbitration can be a transaction or transfer. The transaction-based arbiter multiplexes the data transfer based on the burst transaction. And transfer-based arbiter switches the data transfer based on single transfer.

In this paper, we propose a simplified model of self motivated arbiter, which have the following advantages.

1) It can change the priority policies during run time
2) It is easy to tune the arbitration scheme according to the characteristics of the target application.

The bus network designer uses the communication profile to arrange on-chip bus structures and core connectivity. Its goal is to create a communication network which results in maximum expected throughput. Note that the communication profiler provides only estimates for actual communication delays on any specific network.

On chip bus design has attracted little attention among academic researchers. However there is a number of individual initiatives, mainly within the VLSI Alliance to initiate a set of guidelines for on-chip bus and bus wrapper design. The target of possible standardization is easy of attaching cores with arbitrary bus protocols to system buses. IBM has proposed PLB cross bar switch architecture. Also parallel Inter-module (PI) bus has been proposed to address the demands of real time and fault tolerant application.

II. DEMERITS OF MASTER SIDE ARBITRATION

There are basically three kind of operations, we are accustomed with. They are i) Simple [read / write only at one location at a time], ii)Burst [read /write at multiple location], iii) Out of order [read / write at even or odd location]. There are basically three kind of operations, we are accustomed with. They are i) Simple [read / write only at one location at a time], ii) Burst [read / write at multiple location], iii) Out of order [read / write at even or odd location].

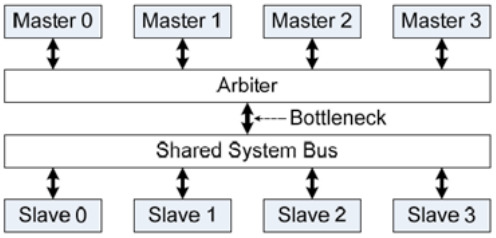

Fig.1. An example of Master side arbitration

Whenever the arbiter go to change its operation between consecutive clock cycles, for example in first clock cycle arbiter is doing simple operation, when in the second clock cycle it will go to burst operation, there will be one clock cycle delay,

which is undesirable. This drawback of Master side arbitration is solved in Slave side arbitration. So the main advantage of slave side arbitration is nothing but minimized latency.

III. DESIGN METHODOLOGY OF ML-AHB BUS MATRIX:

The ML-AHB bus matrix is an interconnection scheme based on the AMBA AHB protocol that allows a number of AHB layers to communicate with a number of AHB shared slaves. Figure [2] shows the overall structure of the ML-AHB bus matrix.

The ML-AHB bus matrix consists of input stage, serial to parallel converter stage, and output stage embedding arbiter. The input stage is responsible for holding the address and control information, when the transfer to a slave is not able to commence immediately. Therefore, there are many flip-flops and multiplexers in the input stage. Another main function of the input stage is to create the slave response signal for the master input port.

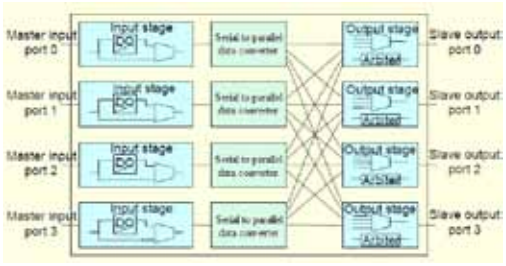

Fig.2. Overall structure of the ML-AHB bus matrix of ARM

The serial to parallel converter block determines which slave a transfer is determined for. The output stage is used to select which of the various master input ports is routed to the slave. Each output stage has an arbiter. This arbiter is used to select a master based on the request provided by the arbiter. In the slave side arbitration, the arbiter is located near slave, which causes the minimization of latency.

Latency is an important attribute when multiple numbers of masters are trying to share a common bus in a system. Minimizing the latency of a particular master is a traditional method. It used to increase the latency of other masters. Masters can change their priority level and can issue the desired transfer length to the arbiters in order to implement a SM arbitration scheme.

In the above example, the serving latency for master 1, 2, 3 span 4, 8 and 2 clock cycles respectively. All the requests are made at the clock cycle 0 and M3 is the most latency sensitive master.

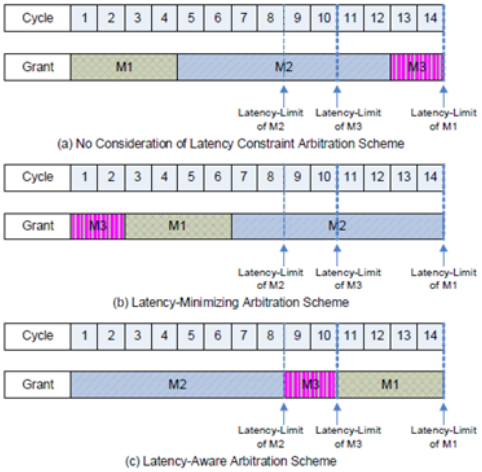

Fig.3. Example of arbitration schemes

A 32-bit address bus is used here, which is applied to the serial to parallel converter. As a result its bits will be divided into 3-bit Slave Number (s_number), 3-bit priority level (p_level), 4-bit Transfer Length (t_length) and 22-bit offset address (offset_add).

Fig.4. Decoding information of 32 bit address bus

Furthermore, these parallel bits are applied into the arbiter to perform the operation. The priority level and transfer length can be changed by the SM demand of each master.

Figure [5] shows the internal architecture of the high performance arbiter, which is based upon the self motivated (SM) arbitration scheme. The arbiter is located near slave side rather than master side in order to minimize the latency. Inside the arbiter, there are Round Robin block, Priority block, Multiplexer, Controller and D Flip-flops. The Round Robin block performs the round robin operation. Similarly the Priority block performs the Priority operation. If all the priority levels are same, then Round Robin operation would be performed, or else Priority operation would be

performed. Initially the up_masked_vector and dn_masked_vector are to found by bitwise AND-ing operation between masked_vector and requested_master.

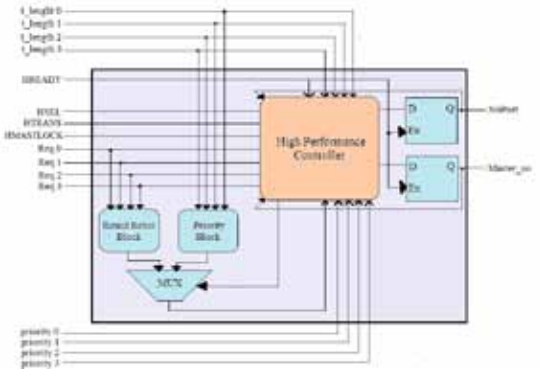

Fig.5. Internal Structure of arbiter

If the up_masked_vector is Zero, then Round Robin operation will be performed on to dn_masked_vector. Otherwise the Round Robin operation ill be performed on to up_masked_vector. The next_master will be updated after one clock cycle.

The internal process of Round Robin block is given in the figure [6].

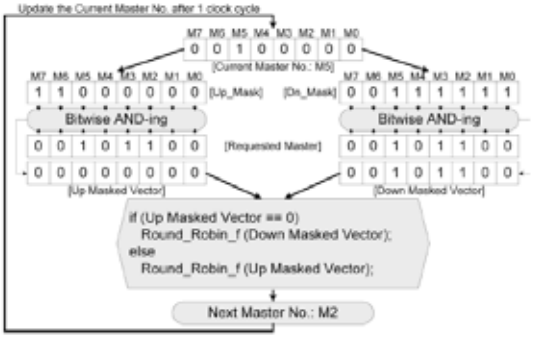

Fig.6. Internal process of the RR block

The Round Robin function operation is also given below.

```
function Round_robin (masked_vector)
variable master_no;
variable present_state, next_state;
variable load;
If(masked_vector ≠ 0) then load = 1;
else load =0;
begin
at the falling edge of clock cycle
for each load = 1
 if (masked_vector (i) = = 1)
 master_no ← i; next_state ← present_state ++;
 else
 master_no ← 2'bz; next_state ← present_state;
end
return master_no;
end
```

Fig. 7. Internal process of Round Robin Function

In figure [8], the master with highest priority is selected are to be selected. The Priority Block operation is also given below.

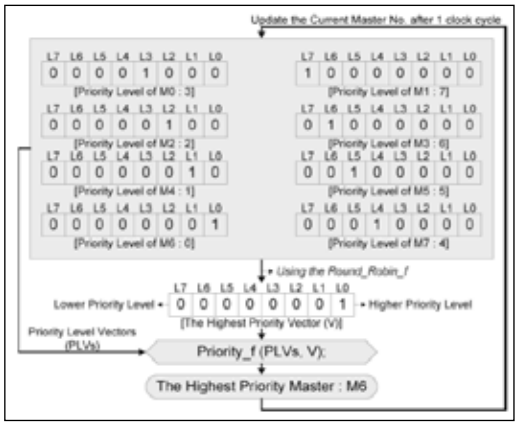

Fig.8. Internal process of Priority block

Also the operation inside the Priority function is also given below, where the highest priority is to be selected.

```
function priority (priority_level, masked_vector)
variable master_no;
variable present_state, next_state;
if (priority_level ≠ 0) then
load = 1;
else
load = 0;
begin
at falling edge of clock cycle
for each load = 1
 if (masked_vector(i) == 1) then
 master_no ← priority level i;
 next_state←present_state ++;
 else
 master_no ← 2'bz;
 next_state ← present_state;
end
return master_no;
end
```

Fig.9. Internal process of Priority Function

The controller is used to compare the priority level of the requesting masters. If the masters have equal priorities, the Round Robin operation would be performed. Or else Priority operation would be performed. The controller also makes the final decision on the master for the next master based on the transfer length of the selected masters.

The control process follows the following three steps.

1) If HMASTLOCK is asserted, the same master remains selected.

2) If HMASTLOCK is not asserted and the currently selected master does not exit, the following hold.

a) If no master is requesting access, the NoPort signal is asserted.

b) Otherwise, a new master for the next transfer is initially selected. If the masters have equal priorities, the Round Robin arbitration scheme is selected; otherwise, the priority arbitration scheme is chosen. In addition, the counter is updated based on the transfer length of the selected master.

3) If none of the previous statements applies, the following hold.

a) If the inbuilt counter inside the arbiter is expired, the following hold.

i) If the requesting masters do not exit, the NoPort signal is updated based on the HSEL signal is "1", the same master remains selected, and NoPort signal is deasserted. Otherwise NoPort signal is asserted.

ii) Otherwise, a master for the next transfer is selected based on the priority levels of the requesting masters. Also the counter inside the arbiter is updated.

b) If the counter is not expired, and the HSEL signal of the master is "1", the same master remains selected, and the counter is decreased.

If the requesting masters do not exit, the NoPort signal is asserted.

The operation inside the SM controller is given below.

```
function controller (equ_priority, Hsel, no_port,
master_no, cont)
at positive edge of clk
return priority;
at negative edge of clk
if (master_no (i) == (i))
 begin
Hsel (i) ← 1;
Add_out ← offset_add(i);
end
at negative edge clk
if (c(i)_en ==1)
begin
if (cnt(i) == t-length(i))
begin
Hsel (i) ← 0;
Cont (i) ← 0;
end
end
```

Fig.10. Internal Process of SM Controller

The SM arbitration scheme is achieved through iteration of the aforementioned steps. Combining the priority level and desired transfer length of the masters allow our arbiter to handle the transfer-based fixed-priority, round-robin and dynamic priority arbitration schemes (abbreviated as FT, RT and DT respectively). Also transaction-based fixed-priority, round-robin and dynamic priority arbitration schemes (abbreviated as FR, RR and DR respectively). Moreover, this arbiter can also deal with desired transfer length based fixed-priority, round-robin and dynamic priority arbitration schemes (abbreviated as DT, DR and DL respectively).

In transfer-based arbitration, the transfer length is allocated as 1, which indicates single transfer. In transaction based arbitration, the transfer length is equal to HBURST signal. In addition, the transfer length for desired transfer length based arbitration is allotted by the demand of each master.

IV. IMPLEMENTATION RESULTS AND PERFORMANCE ANALYSIS

A. Implementation Results

We implement different slave side arbitration schemes for ML-AHB bus matrix. Each arbitration scheme based bus matrix was implemented with synthesizable Verilog. The Xilinx design tool is used to measure the tool area. The implemented arbitration schemes are as follows: FT, FR, RT, RR, DT, DR and self-motivated (SM) arbitration schemes.

The output waveforms of each module of the ML-AHB bus matrix are shown below.

Fig.11. Output waveform of D flip-flop

Fig.12. Output waveform of Multiplexer

Fig.13. Output waveform of Input Stage

The output of second block / Serial to Parallel converter Is used to split the 32 bit address into 3-bit slave number, 3-bit priority level, 4-bit transfer length and 22-bit offset address.

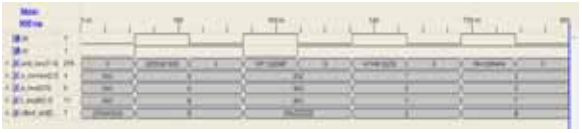

Fig.14. Output waveform of Serial to Parallel converter

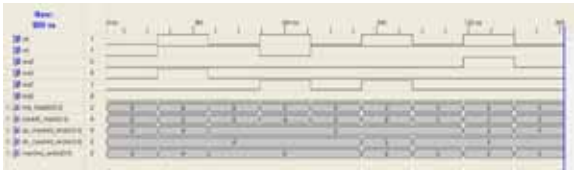

Fig.15. Output waveform for masked vector generation in RR block

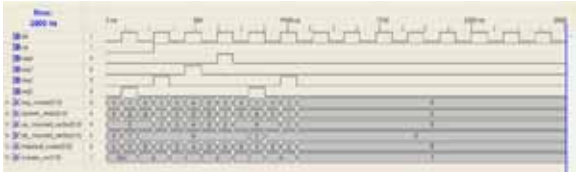

Fig.16. Output of Round Robin (RR) block

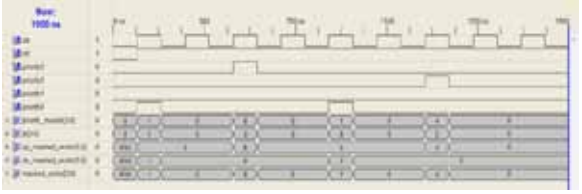

Fig.17. Output waveform for masked vector generation in P block

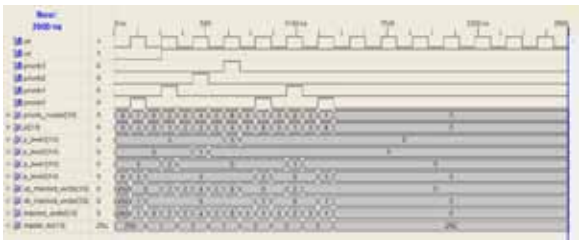

Fig.18. Output of Priority (P) block

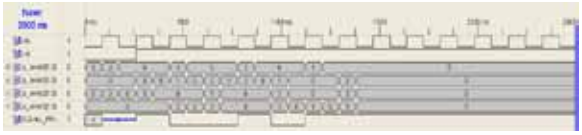

Fig.19. Output of Equal priority (SM controller)

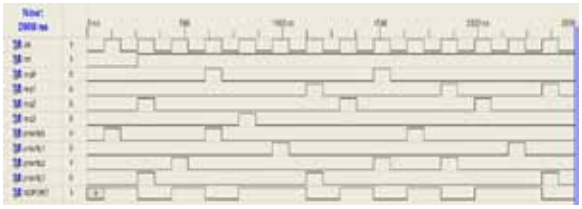

Fig.20. Output of NoPort Pin (SM controller)

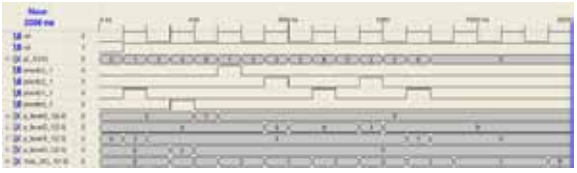

Fig.21. FT arbitration scheme

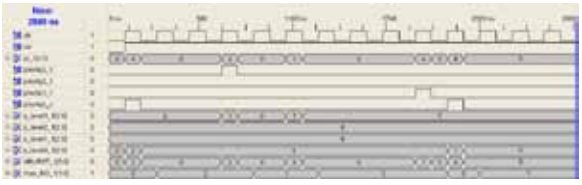

Fig. 22. FR arbitration scheme

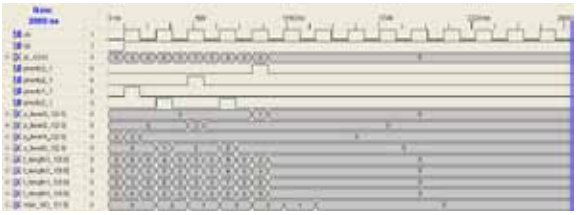

Fig.23. FL arbitration scheme

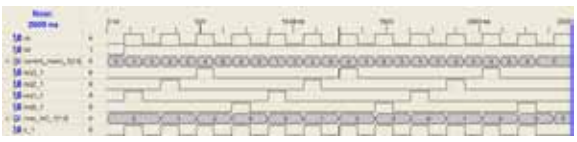

Fig.24. RT arbitration scheme

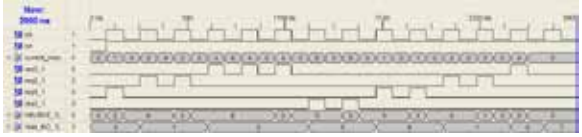

Fig.25. RR arbitration scheme

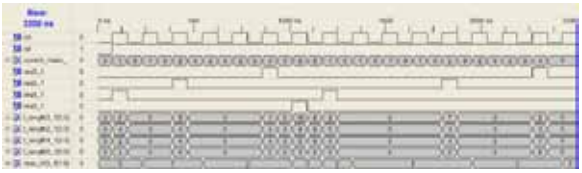

Fig.26. RL arbitration scheme

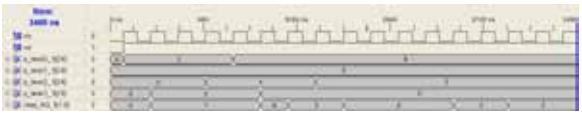

Fig.27. DT arbitration scheme

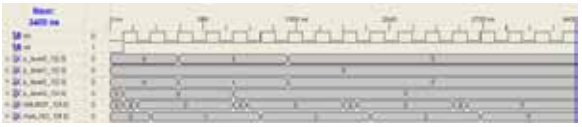

Fig. 28. DR arbitration scheme

CONCLUSION

In this paper, a flexible arbiter is proposed which can change the arbitration scheme during the run time. It has nine arbitration schemes, out of which one will be selected at any particular time instant. Experimentally it can be proved that even though the area overhead will be little bit increased which is undesirable, still the overall throughput increases in much higher ratio.

For further work, we feel that the configuration of the SM arbitration scheme with the maximum throughput need to be found automatically during runtime. We are likewise looking at the applicability of the proposed arbitration scheme to the AMBA AXI.

REFERENCES

[1] M. Drinic, D. Kirovski, S. Megerian, and M guided on-chip bus-network design," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 25, no. 12, pp. 2663–2673, Dec. 2006. [2] S. Y. Hwang, K. S. Jhang, H. J. Park, Y. H. Bae, and H. J. Cho, "An ameliorated design method of ML-AHB bus matrix," ETRI J., vol. 28, no. 3, pp. 397–400, Jun. 2006. [3] S. Y. Hwang, H. J. Park, and K. S. Jhang, An Efficient Implementation Method of Arbiter for the ML-AHB Busmatrix. Berlin, Germany: Springer-Verlag, May 2007, vol. 4523, LNCS, pp. 229–240. [4] J. Turner and N. Yamanaka, "Architectural choices in large scale ATM switches," IEICE Trans. Commun., vol. E-81B, no. 2, pp. 120–137, Feb. 1998. [5] S. Y. Hwang, H.-J. Park, and K.-S. Jhang, "Performance analysis of slave-side arbitration schemes for the multi-layer AHB busmatrix," J. KISS, Comput. Syst. Theory, vol. 34, no. 5, pp. 257–266, Jun. 2007. [6] E.-G. Jeong, J.-G. Lee, K.-S. Jhang, J.-A. Lee, and D. Har, "Asynchronous layered interface of multimedia socs for multiple outstanding transactions," J. VLSI Signal Process. Syst., vol. 46, no. 2/3, pp. 133–151, Mar. 2007. [7] K. Lahiri, A. Raghunathan, and S. Dey, "Performance analysis of systems with multi-channel communication architectures," in Proc. Int. Conf. VLSI Design, Jan. 2000, pp. 530–537. [8] M. Jun, K. Bang, H.-J. Lee, N. Chang, and E.-Y. Chung, "Slack-based bus arbitration scheme for soft real-time constrained embedded systems," in Proc. Int. Conf. ASP-DAC, Jan. 2007, pp. 159–164.

Sara Publishing Academy
Indian Journal Of Applied Research
Journal for All Subjects

Editor,
Indian Journal Of Applied Research
8-A, Banans, Opp. SLU Girls College,
New Congres Bhavan, Paldi, Ahmedabad-380006.
Contact.: +91-9824097643 E-mail : editor@ijar.in

Printed at Unique Offset, Novatsing Rupam Estate, Opp. Abhay Estate, Tavdipura, Shahibaug, Ahmedabad