An Ameliorated Design Method of ML-AHB BusMatrix

Soo Yun Hwang, Kyoung Sun Jhang, Hyeong Jun Park, Young Hwan Bae, and Han Jin Cho

ABSTRACT—The multi-layer advanced high-performance bus (ML-AHB) BusMatrix proposed by ARM is an excellent architecture for applying embedded systems with low power. However, there is one clock cycle delay for each master in the ML-AHB BusMatrix of the advanced microcontroller bus architecture (AMBA) design kit (ADK) whenever a master starts new transactions or changes the slave layers. In this letter, we propose an improved design method to remove the one clock cycle delay in the ML-AHB BusMatrix of an ADK. We also remarkably reduce the total area and power consumption of the ML-AHB BusMatrix of an ADK with the elimination of the heavy input stages.

Keywords—System-on-a-chip, on-chip bus, ML-AHB BusMatrix, arbitration scheme.

I. Introduction

Recent system-on-a-chip (SoC) designs require a system bus with high bandwidth to perform multiple operations in parallel. To solve the bandwidth problems, there are several types of high performance on-chip-bus proposals such as the multi-layer advanced high-performance bus (ML-AHB) BusMatrix from ARM [1], [2], the processor local bus (PLB) crossbar switch from IBM, CONMAX from Silicore, silicon backplane from Sonics, Inc., and so on. In particular, the advanced microcontroller bus architecture (AMBA) of ARM is conventionally used in a number of systems since it is a good architecture for applying embedded systems with low power [3]. Also, many researches related with the AMBA bus have been proposed, such as multi-clock operations [4], a wrapper design [5], multi-processor design [6], and system level modeling [7]. Moreover, the ML-AHB BusMatrix allows a number of masters to communicate with a number of slaves in a system. The advantage of using the BusMatrix is that it provides parallel access paths between the various masters and slaves, giving an improved overall system bandwidth. However, because of the input stage and arbitration logic realized with a Moore type finite state machine (FSM), there is a one clock cycle delay for each master in the ML-AHB BusMatrix of an AMBA design kit (ADK) whenever the master restarts the transactions or changes the slave layers. In this letter, we present one way to improve the ML-AHB BusMatrix structure of an ADK. With the removal of the input stage and some restrictions on the arbitration scheme, we were able to not only take away the one clock cycle delay but also decrease the total area and power consumption. In the next section we explain the characteristics of the ML-AHB BusMatrix of an ADK including the one clock cycle delay. Section III describes a design method to improve the ML-AHB BusMatrix structure of an ADK. We present our experimental results in section IV, and we summarize this letter and suggest future works in section V.

II. The Characteristics of the ML-AHB BusMatrix of an ADK

Figure 1 shows the overall structure of the ML-AHB BusMatrix of an ADK. The ML-AHB BusMatrix consists of an input stage, decoder, and output stage embedded output arbiter. In the ML-AHB BusMatrix of an ADK, the master starts only the first transfer of the transaction, and then waits for the slave response to proceed to the next transfer. At this point in time, a one clock cycle delay occurs since the transmitted address and control information are stored in the registers of the input stage, and the master select signal of the output arbiter is transmitted at the next clock cycle because of the arbitration logic realized with a Moore type FSM. Figure 2 illustrates the one clock cycle delay that occurs when the master starts a 4-beat incrementing burst transaction.

Manuscript received July 21, 2005; revised Feb. 22, 2006.

Soo Yun Hwang (phone: +82 42 860 5569, email: syhwang@etri.re.kr) and Hyeong Jun Park (email: parkhj@etri.re.kr) are with Mobile Telecommunication Research Division, ETRI, Daejeon, Korea.

Kyoung Sun Jhang (phone: +82 42 821 6653, email: sun@cnu.ac.kr) is with Department of Computer Engineering, Chungnam National University, Daejeon, Korea.

Young Hwan Bae (email: yhbae@etri.re.kr) and Han Jin Cho (email: hjcho@etri.re.kr) are with IT Convergence & Components Laboratory, ETRI, Daejeon, Korea.

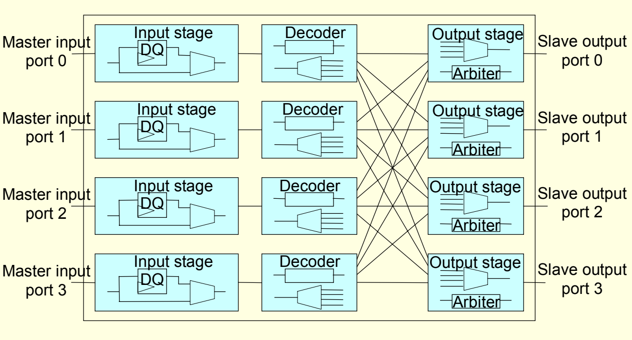

Fig. 1. The overall structure of the ML-AHB BusMatrix of an ADK [2].

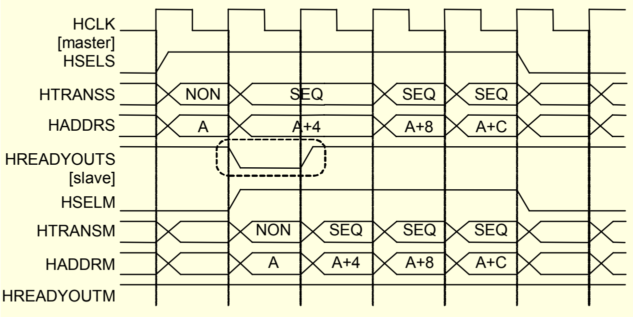

Fig. 2. The occurrence of a one clock cycle delay.

III. The Design Method to Improve the ML-AHB BusMatrix Structure of an ADK

We loosen a part of the AMBA AHB protocol to resolve the one clock cycle delay of the ML-AHB BusMatrix of an ADK. The adjusted protocol allows that the response of a slave to the previous IDLE transfer of a master can be a DELAY as well as an OKAY when the master starts a new transaction. The modification may be valid because the IDLE transfer of the master should be ignored by the slave, and the responsibility of the master is to not initiate a new transfer until it gets an OKAY response in the current clock cycle [1].

In the improved ML-AHB BusMatrix structure as shown in Fig. 3, the input stages are removed and the arbitration logic is implemented with a Mealy type FSM. In our improved ML-AHB BusMatrix, the master has to observe an OKAY response of the slave in order to start a transfer since, although the master sends an IDLE transfer, it may get a DELAY response. The slave response signals are generated by the decoder based on the master select signal of the output arbiter. The master that is just selected can commence the transfer, and the other masters must wait because they receive a DELAY response. Figure 4 shows the internal structure of our improved decoder.

In Fig. 4, the shaded blocks carry out a portion of the main functions of the input stage of the ADK, such as generating response signals. When the master select signal (active) from the output stage is HIGH, the response signals to the master input ports are the same as the equivalent signals from the output stage. When it is LOW, the improved decoder generates a DELAY response (ReadyOut is low and the Resp is OKAY) to stall the transfer. The functions of the other blocks are the same as in the decoder of the ADK, and the meaning of the signals in Fig. 4 is explained in section 2.2 of [1]. We also implemented the output arbiter with a Mealy type FSM. The improved arbiter basically employs a round-robin arbitration scheme based on a masking mechanism [8]. However, there was a problem when we applied the existing preemptive round-robin arbitration scheme to the improved ML-AHB BusMatrix. Figure 5 shows a typical master handover from master 0 to master 1. Master 0 starts a transaction at T1, and master 1 requests a transaction at T3. Master 1 has to wait until master 0 completes a transaction. This is controlled by sending a DELAY response to master 1.

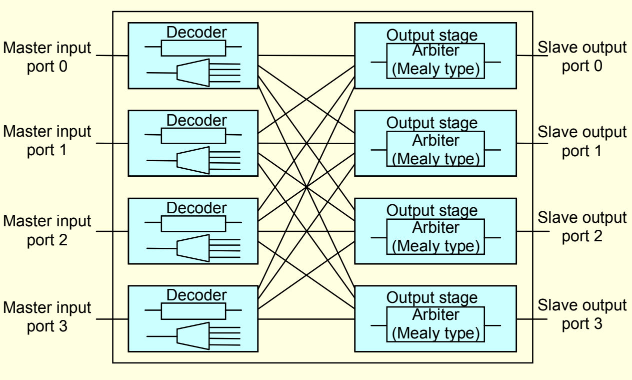

Fig. 3. The overall structure of the improved ML-AHB BusMatrix.

A preemptive master handover from master 0 to master 1 at around T5 makes master 0 lose the first transfer of the next transaction. One way not to lose the transfer is to employ the input stage as in the BusMatrix of the ADK. However, we employ the non-preemptive round-robin arbitration scheme to avoid such a problem and to reduce the area overhead and power consumption of the input stages. In the non-preemptive scheme, the generation of the master select signal depends on the transfer request signal (Sel in Fig. 4) of each master. Figure 6 shows the Mealy type FSM of the improved output arbiter based on the Sel signal.

If the selected master (Current_Master) simply wants a new transaction after the completion of the previous transaction, the master needs to assert a Sel signal, and the arbiter gives the right to use the bus to the selected master. In the non-preemptive round-robin arbitration scheme, starvation may occur since the other masters must wait until all the transactions of the selected master are finished. It may be impossible to

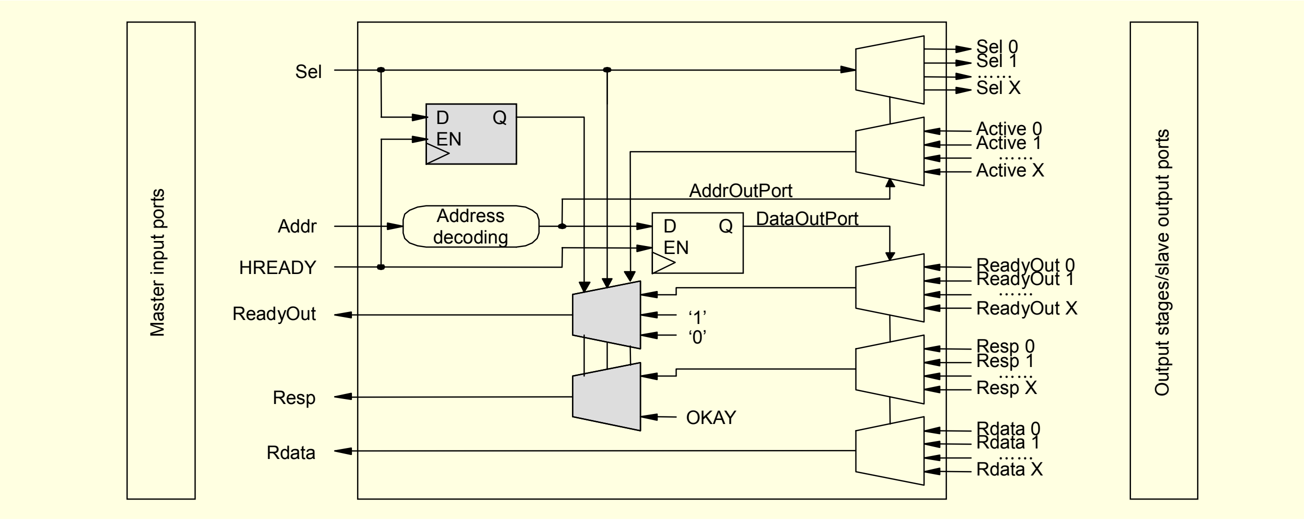

Fig. 4. The internal structure of the improved decoder.

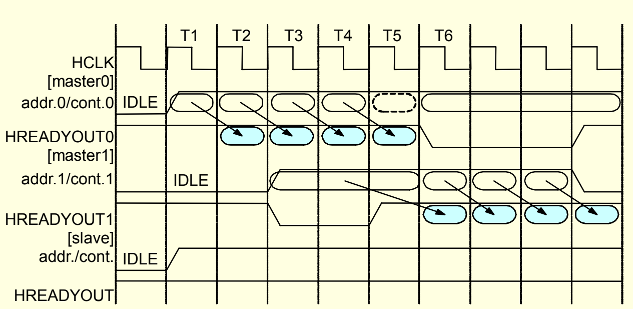

Fig. 5. Preemptive master handover that causes a loss of transfer.

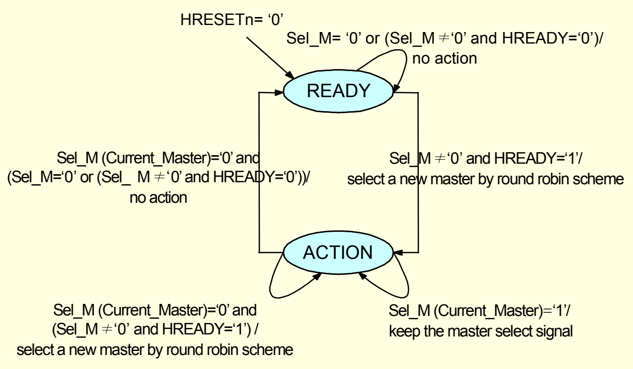
HRESETn= '0'
Sel_M= '0' or (Sel_M ≠ '0' and HREADY='0')/
no action
READY
Sel_M (Current_Master)='0' and
(Sel_M='0' or (Sel_ M ≠ '0' and HREADY='0'))/
no action
Sel_M ≠ '0' and HREADY='1'/
select a new master by round robin scheme
ACTION
Sel_M (Current_Master)='0' and
(Sel_M ≠ '0' and HREADY='1') /
select a new master by round robin scheme
Sel_M (Current_Master)='1'/
keep the master select signal

Fig. 6. Mealy type FSM of the improved output arbiter.

solve the starvation problem because of the AMBA AHB protocol based on a pipelined operation [1]. We may overcome this problem by the insertion of at least one IDLE transfer whenever the selected master completes each transaction. Figure 7 illustrates that master 0 issues one IDLE transfer at T5 so that master 1 or another master is selected. As a result, the starvation problem may be alleviated with the help of master agent modules inserting one or more IDLE transfers after each transaction.

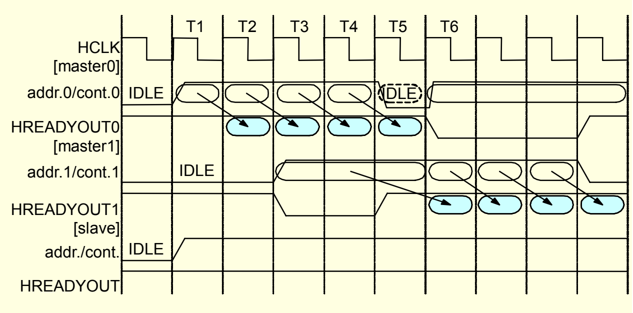

Fig. 7. Timing diagram to show the insertion of one IDLE transfer.

IV. Experiments and Analysis

Figure 8 shows the timing diagram when two masters request a 4-beat incrementing burst transaction to a slave in parallel. Master A started a transaction without a one clock cycle delay, and master B began a transaction after the completion of the transaction of master A. We conducted the performance simulations to compare the improved BusMatrix with the BusMatrix of the ADK. As a result of the performance simulations, we could formulate the improvement (in a percentage) of our approach over the ML-AHB BusMatrix of an ADK with the following expression when considering only one clock cycle delay of the BusMatrix: 1/(Burst Length+1) × 100.

We consider that there is some degradation of the improvement when the improved BusMatrix is used in a real SoC design because of the original delay of each integrated IP and master agent module inserting one or more IDLE transfers after each transaction to avoid starvation.

The BusMatrix was implemented with a synthesizable register transfer level very-high-speed-integrated-circuit hardware description language targeting an XILINX field programmable gate array (XCV 3000), and we used the XILINX design tool

(ISE 7.1i) to measure the total area and clock period. We also used the power compiler of SYNOPSYS to estimate the power consumption. Table 1 shows the experimental results.

There are many registers and 2-to-1 multiplexers in an input stage of the BusMatrix of an ADK. Besides, most of the power in the BusMatrix of an ADK is consumed by the input stages. Table 2 shows the analysis results of the internal parts of the BusMatrix of an ADK on area, delay, and power consumption.

In our improved BusMatrix, the input stage is removed. As a result, the total area, clock period, average static power, and dynamic power consumption are reduced by 33, 33, 64, and 42%, respectively, compared with those of the BusMatrix of an ADK. In particular, though the arbitration logic is implemented with a Mealy type FSM, the clock period of the proposed design is decreased compared with that of the BusMatrix of an ADK. The reason is that the input stages in the BusMatrix of an ADK occupy the largest fraction of the delay determining the clock period among three components, as shown in Table 2, and they are removed in the proposed design.

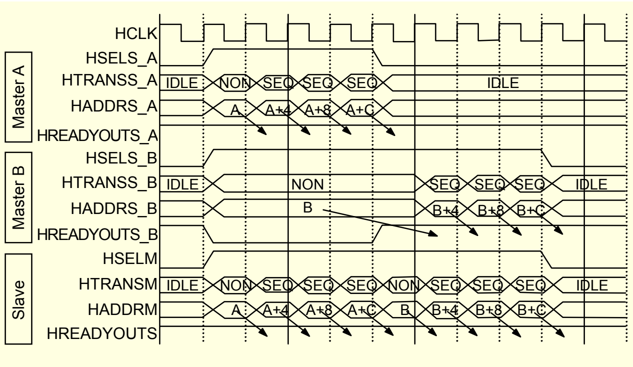

Fig. 8. Timing diagram of the improved ML-AHB BusMatrix.

Table 1. Comparisons of the proposed BusMatrix with that of an ADK in area, clock period, and power consumption.

	Master × Slave	BusMatrix of ADK	Improved BusMatrix
Total area (# of slices)	2 × 2	245	153
	4 × 4	1047	543
	6 × 6	2475	1875
	8 × 8	3689	2748
Clock period (ns)	2 × 2	6.556	3.981
	4 × 4	10.044	6.464
	6 × 6	11.948	8.788
	8 × 8	13.505	9.135
Average static power (nW)		347.4043	122.1017
Average dynamic power (mW)		3.3643	1.9203

Table 2. The analysis results of the internal parts of the BusMatrix.

	Input stage	Decoder	Output stage
Area	38%	19%	43%
Delay (clock period)	44%	25%	31%
Power consumption	64%	9%	27%

V. Summary and Future Works

In this letter, we have presented one way to ameliorate the ML-AHB BusMatrix structure of an ADK. With the removal of the input stage, but at the cost of some restrictions on the arbitration scheme, we could take away the one clock cycle delay and reduce the total area, clock period, average static power, and dynamic power consumption by 33, 33, 64, and 42%, respectively, compared with those of the BusMatrix of an ADK. We expect that it would be better to apply our improved BusMatrix to some applications that do not require the time-division multiple accesses to the slaves. Currently, we are adapting our improved ML-AHB BusMatrix to multimedia applications such as a video phone, MPEG-4, and H.264 codec.

References

[1] ARM, *AMBA Specification Rev 2.0*, 1999.

[2] ARM, *Example AMBA System–User Guide,* 1999.

[3] D, Flynn, “AMBA: Enabling Reusable On-chip Designs,” *IEEE Micro*, vol. 17, issue 4, July-Aug. 1997, pp. 20-27.

[4] M. Dubois, Y. Savaria, and G. Bois, “A Generic AHB Bus for Implementing High-Speed Locally Synchronous Islands,” *Proc. IEEE Southeast Con.*, 8-10 April 2005, pp.11-16.

[5] Yu Ming-Yan, Zhang Qing-Li, Wang Jin-xiang, Ye Yi-zheng, and Lai Feng-Chang “The Design of AMBA AHB/VCI Wrapper,” *Proc. 5th Int'l Conf. ASIC,* vol. 1, Oct. 21-24, 2003, pp. 438 -442.

[6] Youngwoo Kim, Kyoung Park, and Myungjoon Kim, “AMBA Based Multiprocessor System,” *Proc. Int'l Symp., System-on-Chip,* Nov. 19-21, 2003, pp. 41-42.

[7] M. Caldari, M. Conti, M. Coppola, S. Curaba, L. Pieralisi, and Turchetti, C., “Transaction-Level Models for AMBA Bus Architecture Using SystemC 2.0,” *Design, Automation and Test in Europe Conf. and Exhibition,* 2003, pp. 26-31.

[8] Sung-Ho Moon and Dan Keun Sung, “High-Performance Variable-Length Packet Scheduling Algorithm for IP Traffic,” *IEEE GLOBECOM '01,* vol. 4, Nov. 2001, pp. 2666–2670.